DISSERTATION

Extremum Seeking for Dead-Zone Compensation

不感帯補償のための極値探索法

Division of Electrical Engineering and Computer Science

Graduate School of Natural Science & Technology

Kanazawa University

Intelligent Systems and Information Mathematics

Student Number : 1123112104

Dessy Novita

Supervisor: **Prof. Shigeru Yamamoto**

DISSERTATION

Extremum Seeking for Dead-Zone Compensation

Division of Electrical Engineering and Computer Science
Graduate School of Natural Science & Technology
Kanazawa University

Intelligent Systems and Information Mathematics

Student Number: 1123112104

Dessy Novita

Supervisor: Prof. **Shigeru YAMAMOTO**

Abstract

In this thesis, a new technique of dead-zone compensation based on extremum seeking is developed for reducing the steady-state vibration motion of unstable systems with an input dead-zone. To optimally accomplish the dead-zone compensation in real time, we extend the existing extremum seeking method to be able to treat the periodic steady-state output.

The unstable system which we are concerned with is a series connection of a linear time-invariant system and a dead-zone nonlinearity. Although the unstable system is usually stabilized by an appropriate output feedback controller, the closed loop system mostly exhibits the steady-state vibration which is caused by the dead-zone nonlinearity. To cancel the dead-zone nonlinearity, its right inverse of the dead-zone nonlinearity can be used as dead-zone compensation. It is however difficult to obtain the exact right inverse in general. Then, by using the extremum seeking we try to find a probable dead-zone compensation parameter in the right inverse. To this end, a sinusoidal perturbation signal is applied to the dead-zone compensation parameter and the cost function evaluating the vibration of the system is automatically minimized according to the gradient of the cost function. The gradient can be estimated by moving average filter which is also called a mean-over-perturbation-period (MOPP) filter and another sinusoidal perturbation signal. Based on the estimated gradient, the dead-zone compensation parameter is adjusted by an optimizer to obtain the optimal performance.

The effectiveness of the proposed method is illustrated by numerical simulations where we use two models of the self-balancing robot which is a commercial product called e-nuvo WHEEL. One is derived from physical equations of the robot. The other is obtained by Multivariable Output Error type State-sPace Closed-Loop (CL-MOESP) subspace model identification from experimental data of the stabilized closed loop system. In the simulations, the dead-zone compensation parameter converges to the optimal value that minimizes the cost function evaluating the vibration of the body angle of the self-balancing robot. In addition, the simulation results show that the cost function quickly decreases and the vibration of the robot body is rapidly eliminated.

Contents

List of Figures

Acknowledgement

Firstly, I would like to express my gratitude to my academic supervisor Prof. Shigeru YAMAMOTO, for three years guidance, suggestion and feedback he gave me how to do research and to write paper, which have been support in achieving my academic goal of a Ph.D. in Control engineering. Thank you very much for your effort, knowledge, motivation and time. I am also grateful for Associate Prof.Osamu KANEKO for his knowledge, motivation and correction in MoCCoS seminar and for thesis committee Prof. Haruhiko KIMURA, Prof. Masato MIYOSHI and Prof. Itaru HATAUE for their suggestions, comments to prepare this dissertation.

I am deepest thank for my parents, Hasnizar Munaf and Dargal for supporting, praying, attention, motivation me in my study and my life. My sister Filia Fenorika, my young brother Erict Ricardo, my old brother Eldorado and Arifin Nur Muhammad thank for supporting and praying for me. Without their love, patience, encouragement and sacrifice, I would not accomplished it.

I would like to grateful all members MoCCoS laboratory for their valuable helps, Mr. Mohd Syakirin bin Ramli and Mr. Herlambang Saputra for discussing, sharing knowledge and control system, my tutors Mr. Kaminishi and Mr. Masaki Tsuji for helping me daily life in Kanazawa. Mr. Daisuke Ura, Mr. Yuji Okano and Mr. Fumiaki Sawakawa for helping and discussing e-nuvo WHEEL experiments. All of my friends in Indonesian students group Kanazawa especially DIKTI KU 1.

I also wish grateful to Indonesian Directorate Higher Education Ministry of Education and Culture (DIKTI) for fellowship and to Padjadjaran University for allowing to continue my Ph.D. study.

Last but not least, I would like to thank for the deepest heart to my God ALLAH S.W.T. Without ALLAH SWT, I can not do everything.

Thank you everything,

Dessy Novita

Kanazawa , June 2014

Chapter 1

Introduction

1.1 Motivations and Objectives

Most actuators have nonlinearities that deteriorate control system performance. One of such typical nonlinearities is an input dead-zone property. The system with input dead-zone is insensitive for small input signals. Dead-zone nonlinearities in actuators causes not only instability since the feedback signal in closed-loop is ruined, but also large overshoot, large setting time and vibration. For example, it can be seen in a self-balancing robot as an inverted pendulum which is desired to be stabilized motion and impedes balancing in both standing and moving then vibration motion occurs.

Many works have been done for dead-zone compensation. The most generally methods are adaptive scheme, e.g., adaptive control [24], the adaptive fuzzy scheme [4], sliding mode control with adaptive fuzzy [3], neural network and fuzzy logic [14] and FRIT [26].

In practical use, real time canceling the dead-zone is important. Therefore, we extend a method to eliminate dead-zone to optimize control performance in real time by using extremum seeking. The motivation of this work is to make automatically tuning dead-zone compensation to cancel dead-zone in real time.

Extremum seeking control (ESC) is an adaptive control method which automatically optimizes an unknown objective function of a performance measure in real time. When we apply extremum seeking control, we do not need to know the detailed relation between the plant dynamics and the objective, but we only observe the performance measure of the plant [8]. Extremum seeking control commonly uses a perturbation signal, a low-pass filter, a high-pass filter and an integrator [2], [19], [23] (for the discrete-time case, see [6], [7], and for multi-variables, [1]). So recently, extremum seeking control is developed to treat periodic steady-state, which uses a moving average filter to estimate a gradient of the cost function, (see [8], [9], [12]) but this extremum seeking control is limited in continuous-time control.

In this dissertation, we propose extremum seeking control by moving average filter in discrete-time for periodic steady-state to tune dead-zone compensation that optimize control performance in real time. Our extremum seeking control is based on the result by Haring *et al*. The method is applied to two models of self-balancing

robot. One is derived from physical equations of self-balancing robot commercial product called e-nuvo WHEEL. The other is obtained by multi-variables Output-Error type State-space Closed-loop subspace model identification (CL-MOESP) from experimental data. This work is the first developing to reject dead-zone by discrete-time extremum seeking for periodic steady-state. We choose extremum seeking control to cancel dead-zone because extremum seeking control is simple in theoretical mathematics (by Taylor expansion), simple in implementation (which is extremum seeking control does not need complicated system, just perturbation signal, filter and optimizer are used), simple to get gradient estimation by modulation and demodulation perturbation signal, fast convergence and robust against change of the cost. Extremum seeking control does not require the details of the cost function to be minimized.

1.2 Overview of Previous and Related Researches

1.2.1 Dead-Zone Rejection

Works to rejection of dead-zone have become an interesting research for the control community for a long time. Adaptive schemes are popular methods to cancel dead-zone which is the pioneered by Tao and Kokotović [24]. Tao and Kokotović successfully developed to eliminate the dead-zone by using adaptive control strategy for plants with unknown dead-zones which two sets of adjustable parameters that are a dead-zone inverse and the other for a linear controller adaptively updated to reduce the tracking error and to ensure boundedness of all closed-loop signals [24]. Bessa *et al*. proposed adaptive fuzzy control [4] and sliding mode control with adaptive fuzzy dead-zone compensation [3]. They developed a rejection dead-zone method by proving the boundedness of all closed-loop signals and the convergence properties of the tracking error for nonlinear systems subject to dead-zone input based on Lyapunov stability theory and Barbalat's lemma. An electro-hydraulic servo-system as an application to reject dead-zone is used for illustrative example [4], [3].

Moreover, neural network and fuzzy logic are used to cancel dead-zone by Jang *et al.* [14]. The fuzzy logic system is applied for classification property and tuning algorithm then the neural network is utilized for function approximation ability and neural network weight to become adaptive the saturation and dead-zone compensation [14]. Rubio *et al*. have researched to eliminate dead-zone by using proportional derivative control with inverse dead-zone for pendulum systems [20]. The other method to cancel dead-zone is fictitious reference iterative tuning (FRIT) to tune inverse dead-zone parameters and controller parameters by using one-shot closed-loop experiment and covariance matrix adaptation evolution (CMA-ES) strategy for optimization process of FRIT [26].

1.2.2 Extremum Seeking

The first proposed of extremum seeking was from the paper of Leblanc 1922 that explains a new method for designing an ingenious engineering to keep maximum

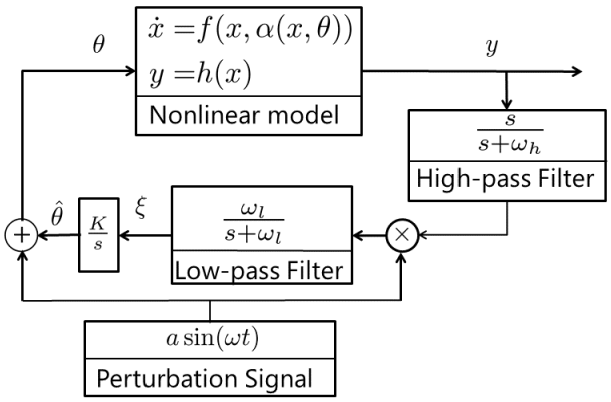


Figure 1.1: Extremum seeking scheme by Krstič [19]

power transfer from a transmission line to a tram car. However, any dynamical model was not used for mathematical analysis and the Leblanc's method is famous for maximizing or minimizing unknown output functions for unknown stable dynamical systems [23], [21]. Many scientists and engineers tackled research on extremum seeking since the stability of the extremum seeking was proved by Krstič and Wang [19],[23]. In the stability analysis for extremum seeking control, averaging technique is utilized for treating the singular perturbation signal. The extremum seeking scheme by Krstič can be shown in Fig. 1.1. The parameter θ is tuned by extremum seeking. The parameter θ is a sum of the signal $\hat{\theta}$ and a slow periodic sinusoidal signal $a \sin \omega t$ as

$$\theta(t) = \hat{\theta}(t) + a \sin \omega t \,. \tag{1.1}$$

The nonlinear model is given by

$$\dot{x}(t) = f(x(t), \alpha(x, \theta)) \tag{1.2}$$
$$y(t) = h(x(t)) \tag{1.3}$$

where x is the state and $\alpha(x, \theta)$ is the feedback control law. Then, the output y comes up to be a static map $y = h(l(\theta))$. Afterward, the output y enters to the high-pass filter $\dfrac{s}{s + \omega_h}$ for clearing the DC component of y.

The output signal of the high-pass filter is multiplied by $a \sin \omega t$. As a result, it produces a signal in phase for $\hat{\theta} < \theta^*$ or out of phase for $\hat{\theta} > \theta^*$. Next, the multiplied signal goes into the low-pass filter which produces a DC component ξ of the signal. The DC componentξ enters into the integrator, and the output $\hat{\theta} = \dfrac{k}{s}\xi$ is used for updating θ until it reaches the optimal value θ^* [19].

The recent research developments in the field of real-time optimization methods for automatic gain tuning is extremum seeking control [2],[6]. Killingsworth and Krstič [15], [16] have developed PID tuning based on extremum seeking control. This method utilizes estimation of parameters by the gradient of the function which is known initial parameters and gotten optimized performance. In Chan *et al.* [5], they have extended a controller to an automatic gain tuning method by modified extremum seeking control and applied to a multi-joint robot. They developed

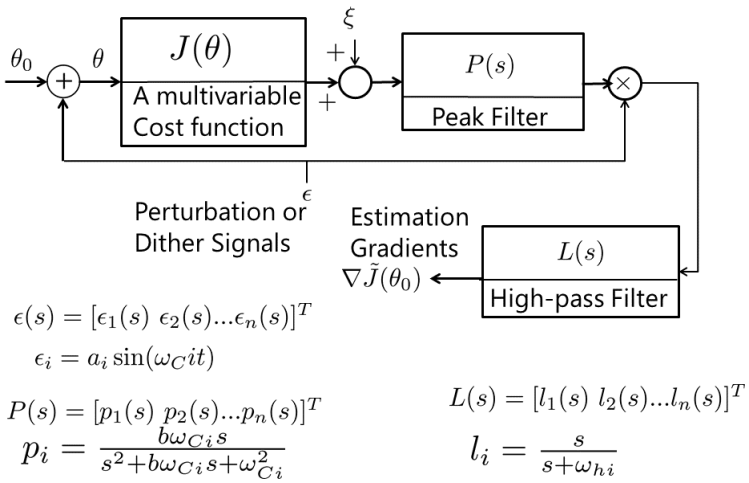


Figure 1.2: Estimation gradient by modified extremum seeking by Kong [18]

a modified extremum seeking control method which is automatic gain tuning by using extremum seeking control with a peak filter to obtain a cost function. This method uses nonlinear programming to minimize the cost function. Modified extremum seeking control has been proposed by Kong *et al*. as in Fig. 1.2 for gradient estimation [5], [18]. This modified extremum seeking control automatically tunes the gain by using a gradient and updating of parameters are optimized until the cost function is minimized to be zero. But, this method still has drawbacks, if it is used for a long time to get results that are not fixed on the gradient. To design extremum seeking control to obtain better performance is gradient estimation that is how to get appropriate filters (low-pass filter, high-pass filter, peak filter, band-pass filter) and perturbation signal parameters then updating parameters of the controller to be optimized. The extremum seeking is different from other optimal control techniques based model free, constant steady-state and also involved explicit knowledge of relation between the parameter and steady state output of the system.

The latest approach to extremum seeking for periodic steady-state was proposed by Haring *et al*. They considered a plant excited by a periodic input disturbances. A periodic steady-state output of the plant has same period with disturbances without explicit knowledge. They designed a cost function which take output information for one period or estimate the output of the plant since measured one period. For stability analysis, semi-global practical asymptotic stability is taken for performance optimization. The extremum seeking for the periodic steady-state can be applied for automated tuning of variable damping for a mass-spring damper system, actuator tuning for (semi-)active suspension systems with periodic disturbance, imbalances identification and compensation in rotary systems, tuning of tracking controllers for repetitive motion tasks of industrial machines such as pick-and place machines, wafer scanner. The extremum seeking for periodic steady-state scheme by Haring is described in Fig. 1.3 [8], [9].

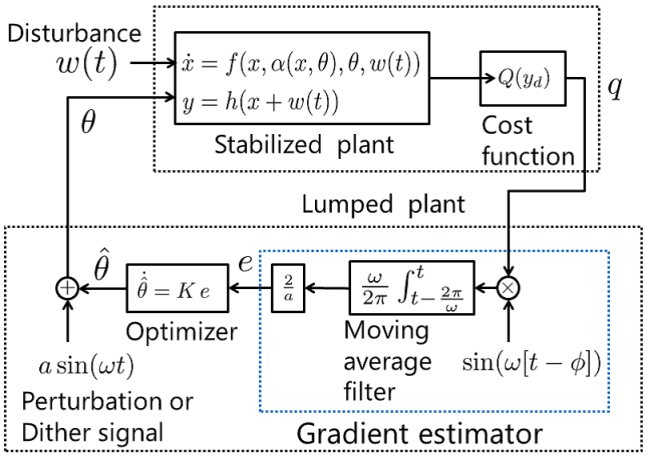


Figure 1.3: Extremum seeking scheme by Haring [8]

1.3 Outline of The Dissertation

This dissertation is organized as follows. Chapter 2 describes a problem formulation that consist of dead-zone, dead-zone compensation and system input with dead-zone.

Chapter 3 explains discrete-time extremum seeking for steady-state together with its convergence, process signal, extremum seeking parameters influencing on performance and stability analysis.

Chapter 4 represents extremum seeking for dead-zone compensation and its application to two models of a self-balancing robot. There are derived from physical equations and multi-variables Output-Error type State-space Closed-loop subspace model identification (CL-MOESP).

In Chapter 5, the works in the thesis are summarized and related future works are mentioned.

Appendix A describes modelling of equations of the self-balancing robot model derived from physical equations. Appendix B represents a procedure of CL-MOESP identification. Appendix C is about stability analysis theorems. Appendix D explains control blocks in simulink and MATLAB programs.

Chapter 2

Dead-Zone Compensation

Dead-zone with a dead-zone interval $[-\delta, \delta]$ ($\delta > 0$) is modelled as a nonlinear function described as

$$D_\delta(u) = \begin{cases} u - \delta & \text{if} \quad u > \delta \\ 0 & \text{if} \quad |u| \leq \delta \\ u + \delta & \text{if} \quad u < -\delta \end{cases} \tag{2.1}$$

The function can be illustrated as in Fig. 2.1. Since dead-zone eliminates small signals which are applied to the system, it causes insensitivity of the system. To eliminate the dead-zone nonlinearity, we introduce another nonlinear function defined by

$$\hat{D}_\delta(\hat{u}) = \begin{cases} \hat{u} + \delta & \text{if} \quad \hat{u} > 0 \\ 0 & \text{if} \quad \hat{u} = 0 \\ u - \delta & \text{if} \quad \hat{u} < 0 \end{cases} \tag{2.2}$$

which is illustrated as in Fig. 2.2. This nonlinear function is right inverse of dead-zone (2.1). That is,

$$D_\delta \circ \hat{D}_\delta = 1, \tag{2.3}$$

in other words,

$$D_\delta(\hat{D}_\delta(u)) = u. \tag{2.4}$$

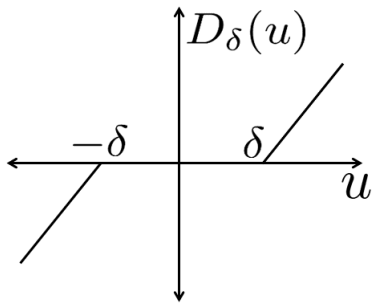


Figure 2.1: Dead-zone

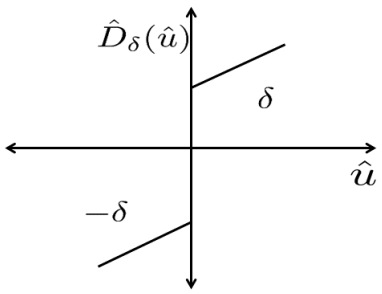


Figure 2.2: Right inverse of dead-zone D_δ

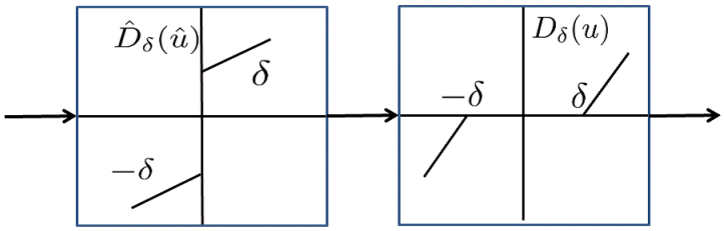


Figure 2.3: Dead-zone compensation

Hence, when we replace $\hat{D}_\delta$ in front of D_δ as in Fig. 2.3, we can cancel the dead-zone.

An example of the input dead-zone can be seen in the mechanism activated by a motor. In this thesis, we will treat the dead-zone in the motor wheel mechanism of a self-balancing robot as modeled an inverted pendulum. In Fig. 2.4, the angle of the body θ of the self-balancing robot is shown for the case where dead-zone is completely cancelled, and it is not cancelled at all. The angle of the body θ exhibits vibration when the input dead-zone in the self-balancing robot is not cancelled.

In this thesis, we consider a single-input and multi-output system which consists of a linear time-invariant part P and an input dead-zone D_δ as shown in (2.1). As in [26], when we know the exact value of δ, we can completely eliminate the dead-zone nonlinearity D_δ by using its right inverse.

We take into consideration a feedback control system to use $\hat{D}_\delta$ as depicted in Fig. 2.6. In the control system, a feedback controller C is designed to stabilize P. In Fig. 2.6, r is the reference input, u is the control input, y is the measured output, respectively. Unlike the ideal case where the exact value of δ is available, it is difficult to cancel D_δ by $\hat{D}_\delta$ completely in practical application. The cancelation error causes the steady-state vibration in the control system when P is unstable. Then, we need to determine an appropriate value δ in $\hat{D}_\delta$ to suppress the steady-state periodic motion in the control system.

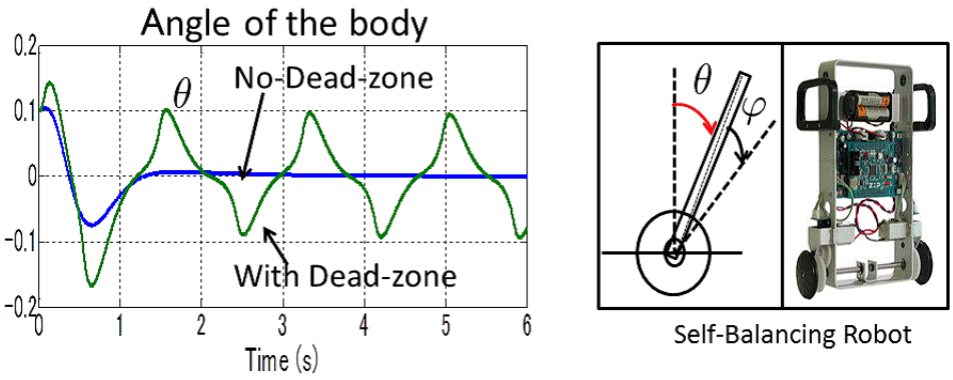


Figure 2.4: The response of the angle of the body of the self-balancing robot when it has dead-zone and no dead-zone

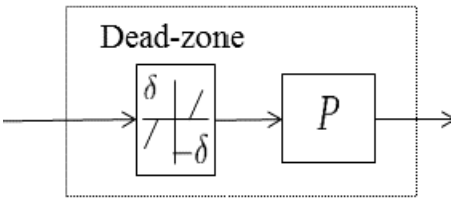


Figure 2.5: A system with an input dead-zone

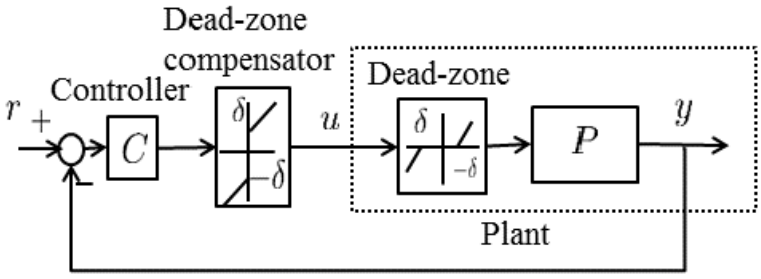


Figure 2.6: Configuration of a feedback control system with dead-zone compensation

Chapter 3

Discrete-time Extremum Seeking Control for Periodic Steady-states

3.1 Discrete-time Extremum Seeking

Extremum seeking control is known as a powerful adaptive method to optimize the control performance in real time. It is mainly used to optimize the control system with a constant steady-state output. In [8], an extremum seeking scheme for periodic steady-state outputs was proposed in the non-equilibrium case. In this thesis, we consider a discrete-time version of [8] which is summarized in Fig. 3.1. The figure shows the configuration of a feedback control system with a tuning parameter δ connected with discrete-time extremum seeking control. We consider a plant as

$$x(k+1) = f(x(k), \delta(k)) \tag{3.1}$$
$$y(k) = h(x(k)). \tag{3.2}$$

The extremum seeking control aims to tune the parameter δ to minimize the cost function of performance output [8] by given as

$$J(\delta(k)) = \left[\frac{1}{N}\sum_{i=k-N}^{k} y(i)^2\right]^{\frac{1}{2}} \tag{3.3}$$

where N is the period of the steady-state output y. The extremum seeking scheme uses a perturbation (dither) signal

$$d_1(k) = a\cos\frac{2\pi}{L}k \tag{3.4}$$

with the period $L \in Z$ and an estimate $\hat{\delta}$ of an optimal value δ^* by applying

$$\delta(k) = \hat{\delta}(k) + d_1(k) \tag{3.5}$$

to the system. We denote the estimation error by

$$\tilde{\delta}(k) = \delta^* - \hat{\delta}(k) \tag{3.6}$$

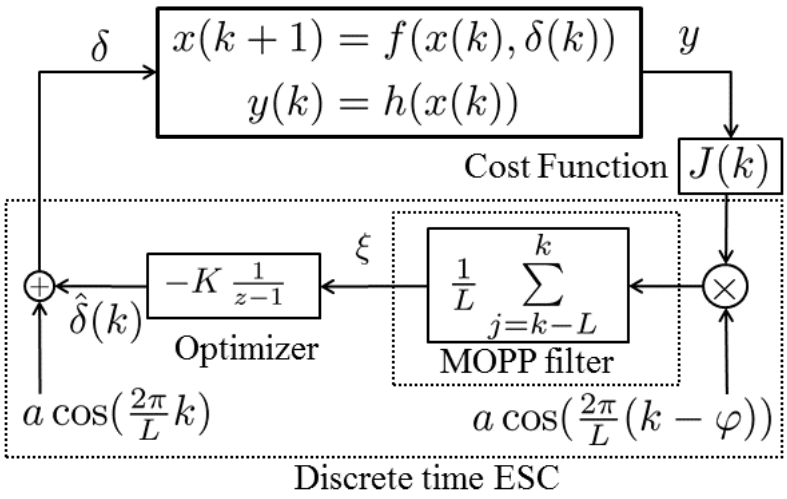


Figure 3.1: Discrete-time ESC scheme

To use (3.5) and (3.6), we have

$$\delta(k) = \delta^* - \tilde{\delta}(k) + d_1(k). \tag{3.7}$$

This perturbed signal affects (3.3). By applying the Taylor series expansion to (3.3), we have

$$\begin{aligned} J(\delta(k)) &= J(\delta^* - \tilde{\delta}(k) + d_1(k)) \\ &= J(\delta^*) + \frac{\partial J}{\partial \delta}(\delta^*)[(\delta^* - \tilde{\delta}(k) + d_1(k)) - \delta^*] + \frac{1}{2}\frac{\partial^2 J}{\partial \delta^2}[(\delta^* - \tilde{\delta}(k) + d_1(k)) - \delta^*]^2 \\ &\cong J(\delta^*) + \frac{\partial J}{\partial \delta}(\delta^*)(d_2(k) - \tilde{\delta}(k) + \frac{1}{2}\frac{\partial^2 J}{\partial \delta^2}(\delta^*)(d_2(k) - \tilde{\delta}(k))^2, \end{aligned} \tag{3.8}$$

where $d_2(k)$ denotes the time delayed signal of $d_1(k)$ due to the dynamics in the closed-loop system as

$$d_2(k) = a\cos\frac{2\pi}{L}(k-\varphi),\ \varphi \in Z. \tag{3.9}$$

Since $J(\delta)$ is optimal at δ^*, $\frac{\partial J}{\partial \delta}(\delta^*) = 0$. Hence,

$$J(\delta(k)) \cong J(\delta^*) + \frac{1}{2}\frac{\partial^2 J}{\partial \delta^2}(\delta^*)(d_2(k) - \tilde{\delta}(k))^2. \tag{3.10}$$

This cost function is multiplied by the demodulation signal $d_2(k)$, and applied into a moving-average filter, also called a mean-over-perturbation-period (MOPP) filter, over the period of $d_2(k)$. Then, the output is

$$\xi(k) = \frac{1}{L}\sum_{j=k-L}^{k} d_2(j)\left(J(\delta^*) + \frac{1}{2}\frac{\partial^2 J}{\partial \delta^2}(\delta^*)(d_2(k) - \tilde{\delta}(k))^2\right). \tag{3.11}$$

By simple calculation, we have

$$\sum_{j=k-L}^{k} d_2(j) = 0, \quad \sum_{j=k-L}^{k} d_2^2(j) = \frac{a^2 L}{2}, \quad \sum_{j=k-L}^{k} d_2^3(j) = 0. \tag{3.12}$$

Hence, when we can assume that $\tilde{\delta}(j)$ is constant over the period L, we have

$$\xi(k) = -\frac{a^2}{2}\frac{\partial^2 J}{\partial \delta^2}(\delta^*)\tilde{\delta}(k). \tag{3.13}$$

The signal $\xi(k)$ is used to generate the estimate $\hat{\delta}$ by using the optimizer (the discrete-time integrator) as

$$\hat{\delta}(k) = -K\frac{1}{z-1}\xi(k). \tag{3.14}$$

Here z is the time-shift operator, that is $z\hat{\delta}(k) = \hat{\delta}(k+1)$. Hence, (3.14) is equivalently

$$\hat{\delta}(k+1) = \hat{\delta}(k) - K\xi(k). \tag{3.15}$$

To use (3.5) and (3.11), we can rewrite (3.15) as

$$\begin{aligned} \tilde{\delta}(k+1) &= \tilde{\delta}(k) + K\xi(k) \\ &= \left(1 - K\frac{a^2}{2}\frac{\partial^2 J}{\partial \delta^2}(\delta^*)\right)\tilde{\delta}(k). \end{aligned} \tag{3.16}$$

Hence, we have next theorem.
Theorem 1.
If

$$\left|1 - K\frac{a^2}{2}\frac{\partial^2 J}{\partial \delta^2}(\delta^*)\right| < 1, \tag{3.17}$$

then an estimate $\hat{\delta}$ converges to the optimal value δ^* by extremum seeking. The convergence rate to the optimal value depends on the amplitude a of the perturbation signal d_1 and d_2, and the gain K of the optimizer. Since the Hessian $\frac{\partial^2 J}{\partial \delta^2}(\delta^*)$ of J is unknown, we should start with small values for a and K to find appropriate values. Moreover, the following underlying assumptions are also required [8], [12].

Assumption 1. For all fixed parameter δ over the range for tuning, the stabilized closed-loop system has a unique globally asymptotically stable steady-state solution with a constant period.

Assumption 2. The cost function $J(\delta)$ has a unique global minimum at δ^* for steady-state performance.

We can illustrate the signals in discrete-time extremum seeking control in Fig. 3.2. Firstly, (a) oscillation can be seen in the measured output signal y of the plant (b)

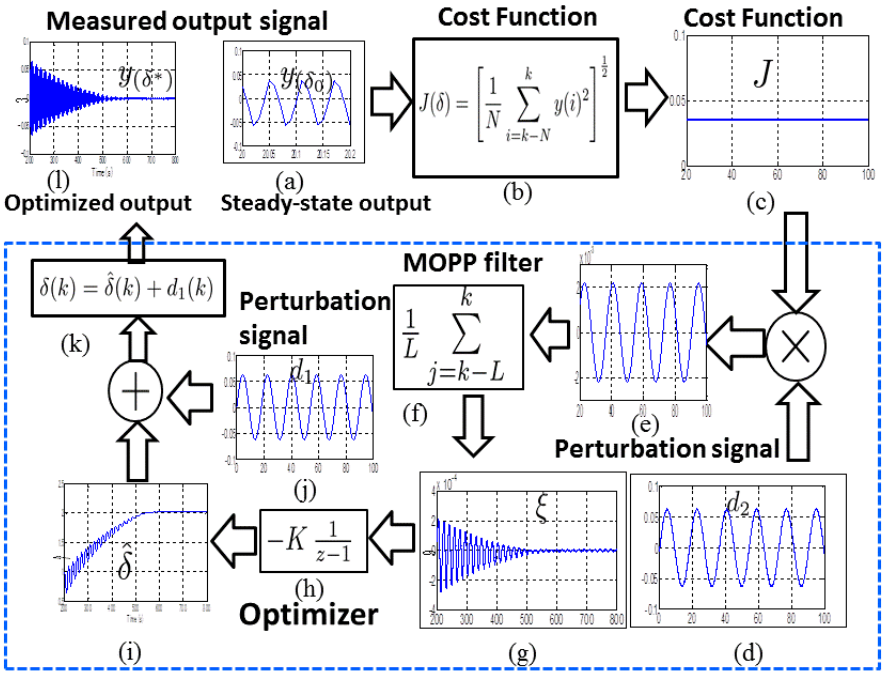


Figure 3.2: Process signals in the discrete-time extremum seeking control

the cost function J (3.3) is calculated to use the output signal y. Secondly, (c) the oscilation in the output y does not appear in the cost function J in the steady-state, then (d) J is multiplied by the demodulation signal d_2 (3.9). Thirdly, (e) the demodulated signal is filtered by (f) the MOPP filter to produce the output signal ξ (g). Futhermore, the output signal ξ gets in (h) the optimizer then the estimation of dead-zone compensation of the signal (i) add with the perturbation signal d_1 (j). Fourthly, the process signal of extremum seeking automatically updates the parameter (k) until we get the optimal value which achieves zero of the output signal of MOPP filter and we can see reducing oscillation in the output measured y signal (ℓ). So, this process indicates effectively and successfully extremum seeking control for automatically tuning parameter to optimize the control performance.

3.2 Extremum Seeking Parameters Influence on Performance

The summarized algorithm for tuning dead-zone compensation by discrete-time extremum seeking consists of
1. Design of a stabilizing controller for the closed-loop system,
2. Design of the cost function of the output measurement of the system,
3. Design parameters of extremum seeking such as frequency or period of perturbation or dither signal, MOPP filter, gain optimizer which check :

- *Period of the output y*
- *Output of the cost function signal with static or constant steady-state*

- *Assumption 1 and 2 are satisfied*
- *Convergence by Theorem 1*

Futhermore, we can design and analyse how to choose properly parameters of extremum seeking to tune dead-zone compensation for good performance and fast rejection dead-zone. The extent of the influence of parameters can be good performance if the parameters of extremum seeking are enlarged and reduced that will be further describe below.

- *Gain of the optimizer K*
  Gain optimizer K adjusts the convergence speed and stability system.
- *Phase of the perturbation signal φ*
  In [8], the phase of the perturbation signal selects the constant $\varphi \in \mathbb{R}_{\geq 0}$ which is an estimate of the sum of the time-varying delay of the plant dynamics and the performance measure of cost function for a good chosen.
- *Period of the perturbation L*
  Period of the perturbation signal L should be chosen larger than period of the cost function N. So, we check the output signal of cost function before designing period of the perturbation signal.
- *Period of MOPP Filter L*
  Period of the MOPP filter is same with the period of the perturbation signal.
- *Amplitude of the perturbation signal a*
  For designing the amplitude of the perturbation signal, we select small value which is smaller than cost function value in initial parameter without extremum seeking.

3.3 Stability analysis

The stability of extremum seeking was first analyzed by Wang and Krstič [19]. They proposed averaging and singular perturbation to derive stability conditions of an extremum seeking feedback scheme [19] in which the averaging theorem adopted theorem 8.3 in Khalil as detail see [17] and Appendix C. To guarantee practical asymptotic stability, Teel et al. [25] proposed a generalized Lyapunov theorem.

Stability analysis of extremum seeking for periodic steady-state suggested by Haring et.al [8]. To apply extremum seeking, we used a stabilized controller which stabilize the plant of system which is a Discrete-Time Linear Quadratic Gaussian (LQG) controller to stabilize the closed-loop system. Stability of the closed loop system is ensured by appropriate state feedback gain and state estimation gain in LQG.

Chapter 4

Extremum Seeking for Dead-zone Compensation and Its Application to a Self-Balancing Robot

4.1 Self-Balancing Robot

In this section, we use discrete-time extremum seeking control discussed in the previous chapter to optimize a dead-zone compensation for the self-balancing robot which is a commercial product called e-nuvo WHEEL shown in Fig. 4.1. The feedback controller C for the self-balancing robot is initially designed to use the model based on dynamic equations and the catalog parameters, and secondly done to use the model obtained by closed loop identification.

4.2 Models of Self-Balancing Robot

4.2.1 Physical equation based model

The equation of motion of the self-balancing robot as an inverted pendulum can be described as

$$\begin{aligned}[(M+m)r_t^2 + J_t + mlr_t\cos\theta(t) + iJ_m]\ddot{\theta}(t) - mlr_t(\sin\theta(t))\dot{\theta}^2(t) \\ +[(M+m)r_t^2 + J_t + i^2J_m]\ddot{\varphi} + c\dot{\varphi} = \eta\, i\, k_t\, u(t).\end{aligned} \tag{4.1}$$

$$\begin{aligned}[(M+m)r_t^2 + J_t + 2mlr_t\cos\theta(t) + ml^2 + J_p + J_m]\ddot{\theta}(t) - mlr_t\sin\theta(t)\dot{\theta}^2(t) \\ -mgl\sin\theta(t) + [(M+m)r_t^2 + J_t + mlr_t\cos\theta(t) + iJ_m]\ddot{\varphi} = 0\end{aligned} \tag{4.2}$$

where physical parameters are defined in Table 4.1. When we assume that

$$\sin\theta(t) \approx \theta,\ \cos\theta(t) \approx 1,\ \dot{\theta}^2(t) \approx 0, \tag{4.3}$$

(4.1) and (4.2) can be rewritten by

$$\begin{aligned}[(M+m)r_t^2 + J_t + iJ_m + mlr_t]\ddot{\theta}(t) \\ +[(M+m)r_t^2 + J_t + i^2J_m]\ddot{\varphi}(t) + c\dot{\varphi}(t) = \eta\, i\, k_t\, u(t)\end{aligned} \tag{4.4}$$

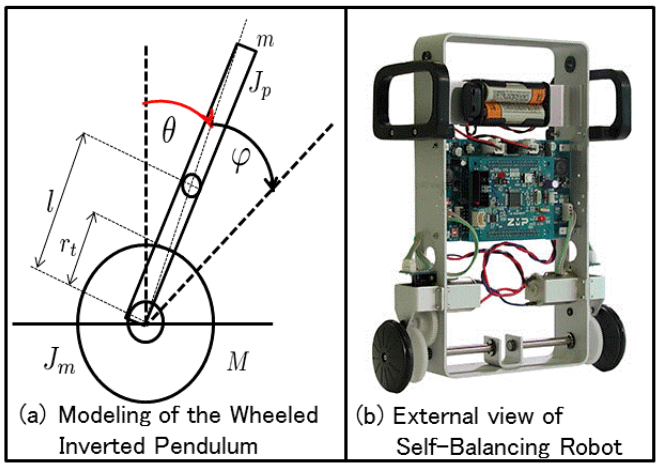


Figure 4.1: Modeling of the Self-balancing robot (ZMP Inc.)

$$\begin{aligned}&[(M+m)r_t^2 + J_t + 2mlr_t + ml^2 + J_p + J_m]\ddot{\theta} \\ &-mgl\theta + [(M+m)r_t^2 + J_t + mlr_t + iJ_m]\ddot{\varphi} = 0\end{aligned} \tag{4.5}$$

As in [27], the state space continuous-time model of the self-balancing robot P in Fig. 4.1 can be derived from physical equations as

$$\dot{x}(t) = A_c x(t) + B_c u(t) \tag{4.6}$$

$$y(t) = C_c x(t) \tag{4.7}$$

where $x = [\theta \ \varphi \ \dot{\theta} \ \dot{\varphi}]^T$ consists of the angle of the body θ, the relative angle of the wheel to the body φ, the angular velocity of the body $\dot{\theta}$ and the relative angular velocity of the wheel to the body $\dot{\varphi}$. The control input u is electrical current. Together with Table 4.1 [27], [28], we have A_c, B_c as

$$\begin{aligned}A_c &= \begin{bmatrix} 0_{2\times2} & I_{2\times2} \\ -E^{-1}G & -E^{-1}F \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 104.05 & 0 & 0 & 0.06 \\ -341.64 & 0 & 0 & -0.37 \end{bmatrix}, \\ B_c &= \begin{bmatrix} 0_{2\times2} \\ -E^{-1}\zeta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 37.8 \\ -232.7 \end{bmatrix},\end{aligned} \tag{4.8}$$

where

$$\begin{aligned}E &= \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix} + \left((M+m)r_t^2 + J_t\right) I_2, \\ F &= \begin{bmatrix} 0 & c \\ 0 & 0 \end{bmatrix}, \ G = \begin{bmatrix} 0 & 0 \\ -mgl & 0 \end{bmatrix}, \ \zeta = \begin{bmatrix} \eta\, i\, K_t \\ 0 \end{bmatrix}\end{aligned} \tag{4.9}$$

$$
\begin{aligned}
e_{11} &= e_{22} = mlr_t + iJ_m \\
e_{12} &= i^2 J_m \\
e_{21} &= 2mlr_t + ml^2 + J_p + J_m
\end{aligned}
$$

Since we measure φ and $\dot{\theta}$,

$$
C_c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.
$$

Table 4.1: Parameters of Self-balancing robot

Mass of the cart (tire, draft shaft ,gear) [Kg]	M	0.071
Mass of the body [Kg]	m	0.5392
Moment of inertia of the body [Kg m^2]	J_p	2.160×10^{-3}
Moment of inertia of the cart [Kg m^2]	J_t	8.632×10^{-5}
Moment of inertia of motor rotor [Kg m^2]	J_m	1.30×10^{-7}
Length between the wheel axle and gravity center of the body[m]	l	0.1073
Radius of the wheel [m]	r_t	0.02485
Friction of the wheel axle [Kg m^2 / s]	c	1×10^{-4}
Torque constant of the motor [N m /A]	K_t	2.79×10^{-3}
Reduction ratio of the gear	i	30
Efficiency drive system	η	0.75

4.2.2 Closed-loop Identification Model

We obtain measurement data of the self-balancing robot commercial product e-nuvo WHEEL and identify the state space model form the measurement data by MOESP-type closed-loop subspace model identification (CL-MOESP) [10], [11]. The obtained model is given by

$$
\begin{aligned}
A &= \begin{bmatrix} 1.0033 & -0.0298 & 0.0157 & -0.0061 & -0.0324 \\ 0.0079 & 0.9102 & -0.2941 & -0.0299 & 0.0013 \\ -0.0030 & 0.1140 & 0.2547 & -0.0970 & 0.0403 \\ -0.0027 & 0.0123 & -0.1012 & 1.0675 & 0.0177 \\ 0.0003 & -0.0023 & 0.2437 & 0.0403 & 0.9341 \end{bmatrix}, \\
B &= \begin{bmatrix} -1.2453 \\ 0.3671 \\ -0.1786 \\ 0.0710 \\ -0.0063 \end{bmatrix}, \quad D = \begin{bmatrix} -0.0108 \\ 0.0401 \end{bmatrix}, \\
C &= \begin{bmatrix} -0.0796 & -0.2888 & -0.0025 & 0.0075 & -0.2407 \\ 0.0148 & -0.2744 & -0.8751 & -0.1124 & 0.2605 \end{bmatrix}.
\end{aligned}
$$

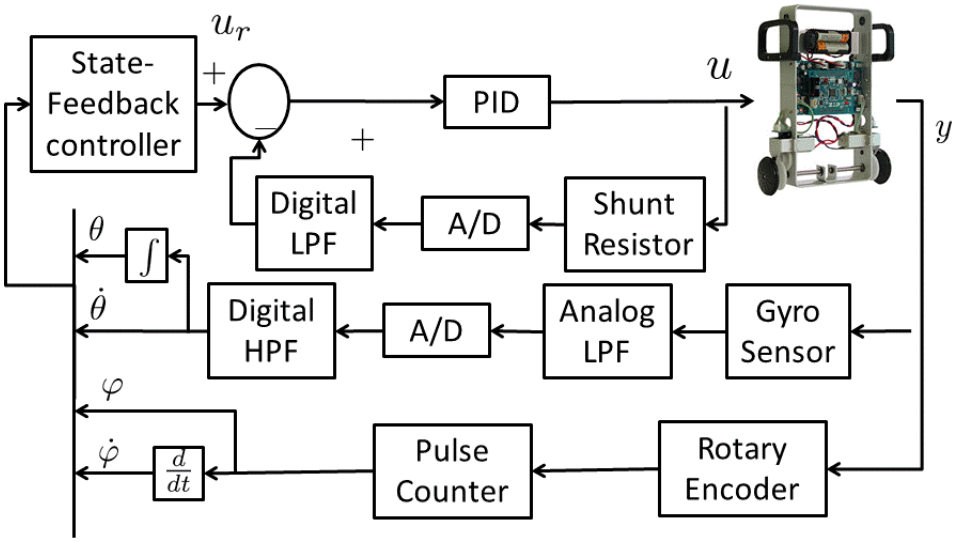


Figure 4.2: Experiment e-nuvo WHEEL scheme

4.3 Design of a Stabilizing Controller

To discretize the continuous-time model (4.1) and (4.2) by zero-order hold, we obtain the discrete-time model

$$x(k+1) = A\,x(k) + B\,u(k) \tag{4.10}$$
$$y(k) = C\,x(k)\,. \tag{4.11}$$

When we use the sampling period $T_s = 0.01$ sec, we have

$$A = \begin{bmatrix} 1 & 0 & 0.01 & 0 \\ -0.02 & 1 & -0.0001 & 0.01 \\ 1.04 & 0 & 1 & 0 \\ -3.42 & 0 & -0.02 & 1 \end{bmatrix}, \; B = \begin{bmatrix} 0.002 \\ -0.01 \\ 0.38 \\ -2.32 \end{bmatrix},$$
$$C = C_c\,.$$

The discrete-time model is used to design the discrete-time LQG controller [13], [22] in Fig. 4.3 which minimizes

$$E\left[\lim_{\tau\to\infty}\frac{1}{\tau}\sum_{k=0}^{\tau} x^T(k)Qx(k) + u^T(k)Ru(k)\right] \tag{4.12}$$

where Q and R are given constant weight matrices for which $Q = Q^T \geq 0$, $R = R^T > 0$, under the existence of the process noise and the measurement noise. We assumed weight matrices $Q = I_4$, $R = 1$ and covariance of the process noise $W = 1$ and the measurement noise $V = 0.012\ I_2$ which means rms noise 1% on each sensor channel. The feedback gain K is derived as

$$K = (B^T S\,B + R)^{-1}B^T S\,A, \tag{4.13}$$

and the solution $S = S^T \geq 0$ of the associated Riccati equation

$$A^T S\,A - S - (A^T S\,B + N)(B^T S\,B + R)^{-1}(B^T S\,A + N^T) + Q = 0\,. \tag{4.14}$$

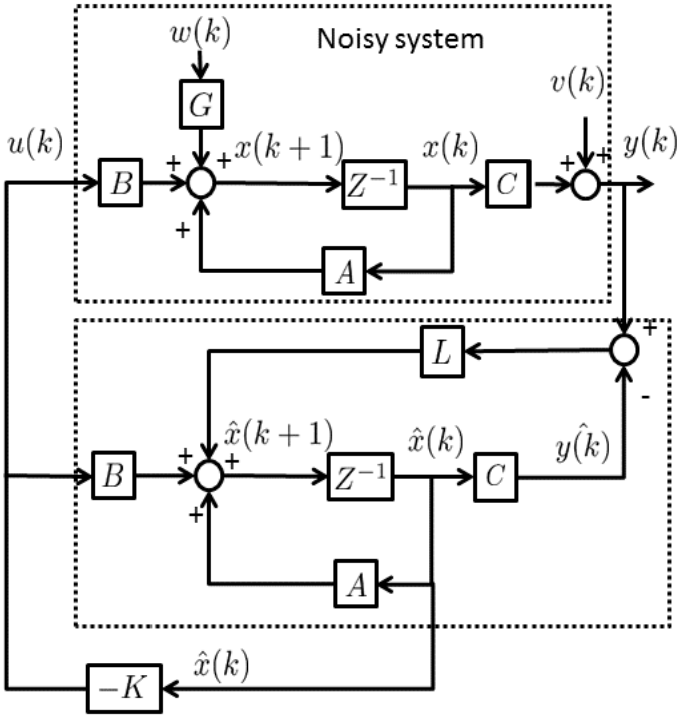


Figure 4.3: Discrete-time Linear Quadratic Gaussian control

The optimal L minimizing $E[x(k) - \hat{x}(k)]^T[x(k) - \hat{x}(k)]$ is given by $L = APC^T(CPC^T + V)^{-1}$ where $P = P^T \geq 0$ is the unique positive-semidefinite solution of discrete algebraic Riccati equation. The discrete-time linear quadratic Gaussian (LQG) controller is given by connecting the discrete-time linear quadratic regulator (LQR) and the discrete-time Kalman filter according to block diagram in Fig. 4.3.

4.4 Physical equation based Model

4.4.1 Dead-Zone Compensation

In the following, we set the actual dead-zone parameter $\delta = 2$ for the numerical simulations. When we do not use the dead-zone compensator (this corresponds to $\hat{\delta} = 0$ in $\hat{D}$, the angle of the body θ shows periodic steady periodic steady-state motion with amplitude 0.1 rad (5.7 degree) as shown in Fig. 4.4 (a). On the other hand, when we use the dead-zone compensator $\hat{D}_\delta$ with $\hat{\delta} = 1$, the amplitude of the periodic steady-state motion of the angle of the body θ is 0.05 rad (2.8 degree) as shown in Fig. 4.4 (b). Although the periodic steady-state motion is much reduced by the dead-zone compensator, it still remains due to the gap between the actual δ and $\hat{\delta}$ in the dead-zone compensator. Hence, it is important to tune $\hat{\delta}$ to suppress the periodic steady-state motion completely.

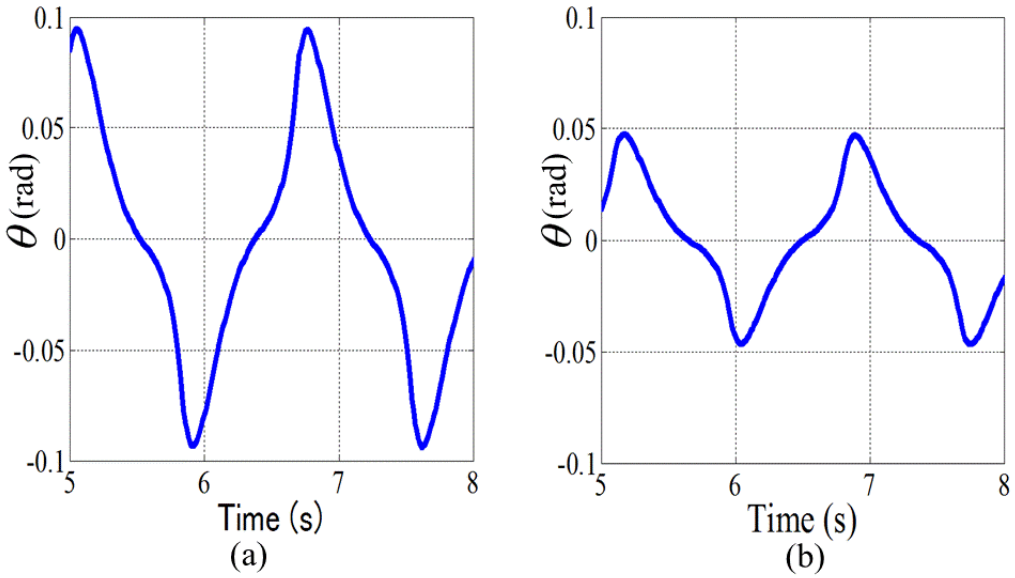


Figure 4.4: The angle of the body θ of the closed-loop system with dead-zone $D_\delta(\delta = 2)$ (a) with no dead-zone compensator, (b) with dead-zone compensator $\hat{D}_\delta(\hat{\delta} = 1)$

4.4.2 Extremum Seeking for Tuning of Dead-Zone Parameter

For simplicity, we use $y = \theta$ for the output for extremum seeking control and the cost function

$$J(k) = \left[\frac{1}{N}\sum_{i=k-N}^{k}\theta(i)^2\right]^{\frac{1}{2}}$$

whereas the output for feedback control is $y = [\varphi \ \dot{\theta}]^T$. The parameters for extremum seeking control are as follows; the amplitude and the period of the perturbation signal are $a = 1/16$ and $L = 1800$, the gain of the optimizer is $K = 3$, the time delay of the perturbation signal in d_2 is $\varphi = 100$, the period of cost function is $N = 180$. The mean-over-perturbation-period can be implemented by a FIR filter. A simulation result where tuning dead-zone compensation by the discrete-time ESC starts at $t = 200$ sec is shown in Fig. 4.5 where $x[0] = [0.01 \ 0 \ 0 \ 0]^T$ as the initial variable. The dead-zone compensation parameter $\hat{\delta}$ converges to $\delta = 2.06$ as shown in Fig. 4.5 (a). Although this final value is not the actual value $\delta = 2$, the periodic steady-state motion in the body θ is sufficiently suppressed as shown in Fig. 4.5 (b). Indeed, the cost function J decreases to sufficiently small value as shown in Fig. 4.5 (c). This result shows that the small gap between the dead-zone parameter compensation and the actual one is acceptable.

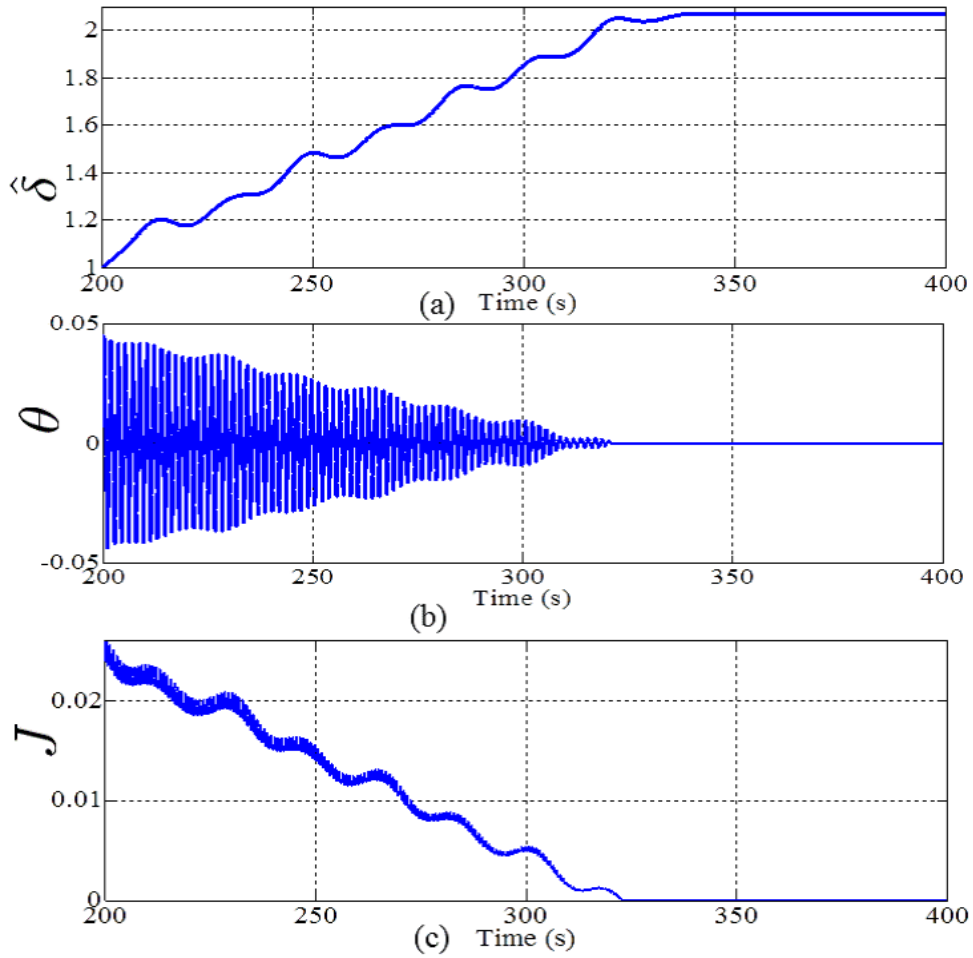


Figure 4.5: A simulation result when extremum seeking is applied for tuning of dead-zone compensator. (a) The tuned value of dead-zone compensator, (b) the angle of the body, (c) the cost function.

4.5 Simulation Results by CL-MOESP Identification Model

4.5.1 Dead-Zone Compensation

We applied the discrete-time LQG regulator to make stabilized unstabled plant from the CL-MOESP identification model of the Self-balancing robot because tuning by the discrete-time ESC was need stabilized plant. We utilized the discrete-time Kalman filter to estimation state and the discrete-time LQR to search state feedback gain. We used weight matrices $Q = I_4, R = 1$ and covariance of process noise $W = 1$ and measurement noise $V = 0.01^2 I_2$ which means rms noise 1% on each sensor channel. Then we set dead-zone parameter $\delta = 2$ and the initial state as by $x_0 = [0.01 \ 0 \ 0 \ 0]^T$. So, we achieved simulation result of the CL-MOESP model of the closed-loop system without extremum seeking with dead-zone, dead-zone compensator and the discrete-time LQG regulator controller. When we utilize dead-zone $D_\delta = 2$ and do not use the dead-zone compensator $\hat{D}_\delta = 0$, the angle of the

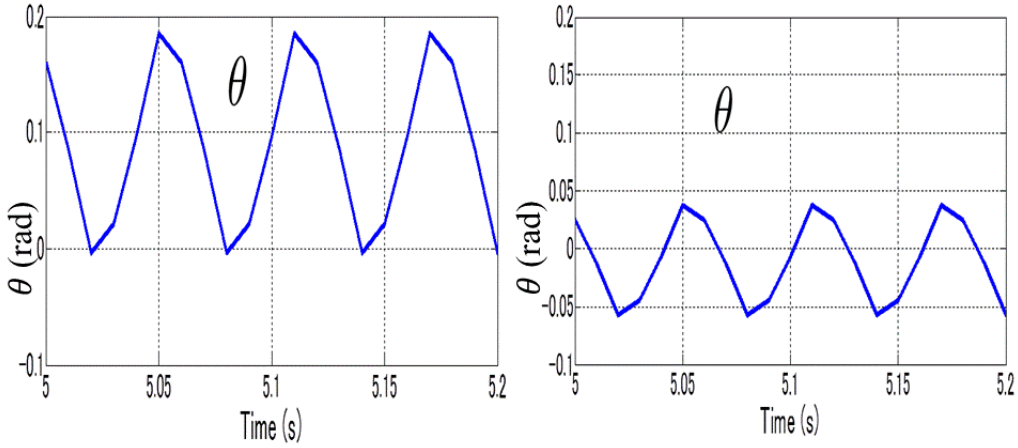


Figure 4.6: The angle of the body θ of CL-MOESP model of the closed-loop system with dead-zone D_δ ($\delta = 2$) (a) with no dead-zone compensator, (b) with dead-zone compensator $\hat{D}_\delta$ ($\hat{\delta} = 1$)

body θ shows periodic steady-state motion with amplitude 0.2 (11.46 degree) rad as shown in Fig. 4.6 (a). When, we use the dead-zone compensator $\hat{D}_\delta$ with $\hat{\delta} = 1$, the amplitude of the periodic steady-state motion of the angle of the body θ reduces to 0.05 rad (2.8 degree) as shown in Fig. 4.6 (b) for the CL-MOESP identification model.

4.5.2 Extremum Seeking for Tuning of Dead-Zone Parameter

We take the discrete-time ESC to tuning dead-zone compensation for rejecting vibration which is cost function from output θ by given

$$J(\delta) = \left[\frac{1}{N} \sum_{i=k-N}^{k} \theta(i)^2 \right]^{\frac{1}{2}} .$$

Afterward, we set extremum seeking parameters that are same with the previous setting of self-balancing robot derived physical equations model. Simulation result of the CL-MOESP model for tuning dead-zone compensation by the discrete-time ESC are shown in Fig. 4.7 with $K = 3$ and in Fig. 4.8 with $K = 10$ which are represented cost function of the CL-MOESP model J in Fig. 4.7 (a) is decrease to minimum, the angle of the body of the CL-MOESP model depict for rejection dead-zone and stabilized moving self-balancing robot in Fig. 4.7 (b), estimation dead-zone compensation $\hat{\delta}$ that is achieved 2 as fit as setting dead-zone it is shown in Fig. 4.7 (c) the tuned value of dead-zone compensator, but it needs time starting 200 second and achieved the optimal value after 500 second for $K = 3$ and after 250 second for $K = 10$. It has fastly convergence to optimal performance for gain optimizer $K = 10$ but it has negative value estimation delta around 210 second. Afterthat, estimation ξ (Fig. 4.7 (d)) is zero. This indicates optimal performance.

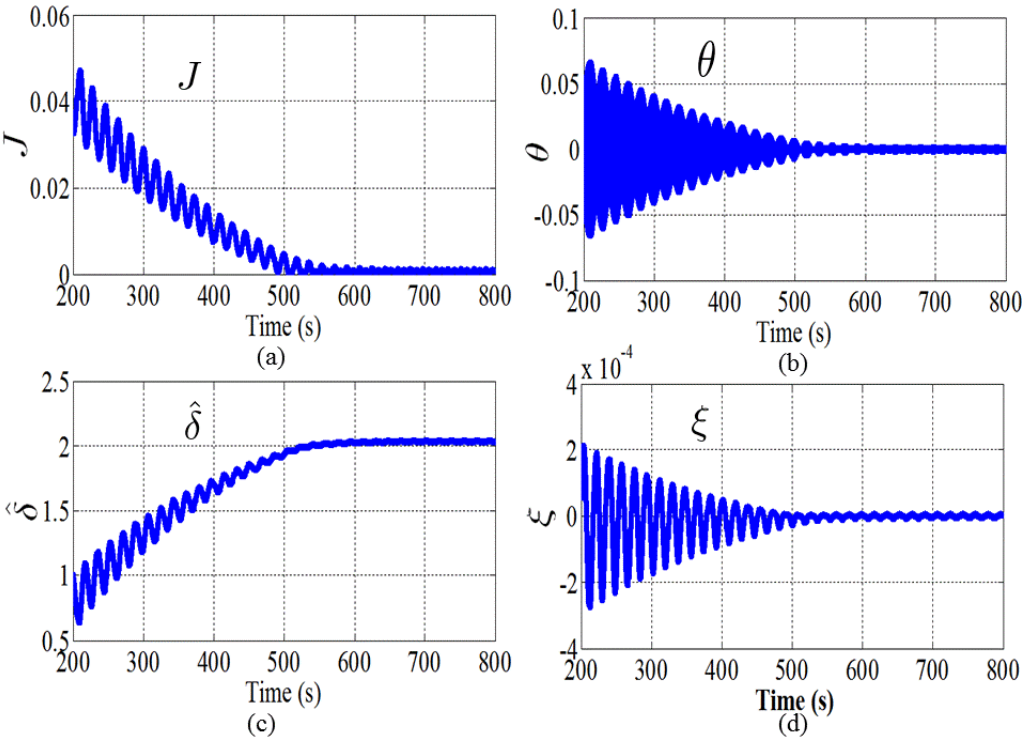


Figure 4.7: Extremum seeking result when $K = 3$ for CL-MOESP model. Time response of (a) cost function (b) the angle of the body (c) tuned parameter (d) estimated gradient of the cost function

4.5.3 Analysis of Extremum Seeking Parameters

In Chapter 3, extremum seeking consists of four parameters which analyze the influence of parameters can be good performance or not if the parameters of extremum seeking are enlarged or reduced that will be further describe below.

- *Optimizer gain K*
  The optimiser gain K influences on the performance of the system as shown in Fig. 4.9 by the self-balancing robot CL-MOESP model. As results when we changed choose $K = 1,\ 3,\ 10,\ 20,$ and 30 we show the cost function J in Fig. 4.10, output θ in Fig. 4.11, estimation delta $\hat{\delta}$ in Fig. 4.12 and ξ in Fig. 4.13. The performance when we use $K = 10$ is fastest to get optimal value about 50 second, but it has negative value around 210 second.

- *Phase of perturbation signal φ*
  We select six phases of perturbation signal φ that are $\varphi = 0.1$, 1, 10, 30, 100 and 200. In Fig. 4.14, we show the simulation results for tuning dead-zone compensation by the CL-MOESP model, there are no significantly differences in the cost function J, estimation delta $\hat{\delta}$, ξ and the angle of the body θ in Fig. 4.15.

- *Period of perturbation signal L*
  Period L of the perturbation signal is chosen larger than period of the cost

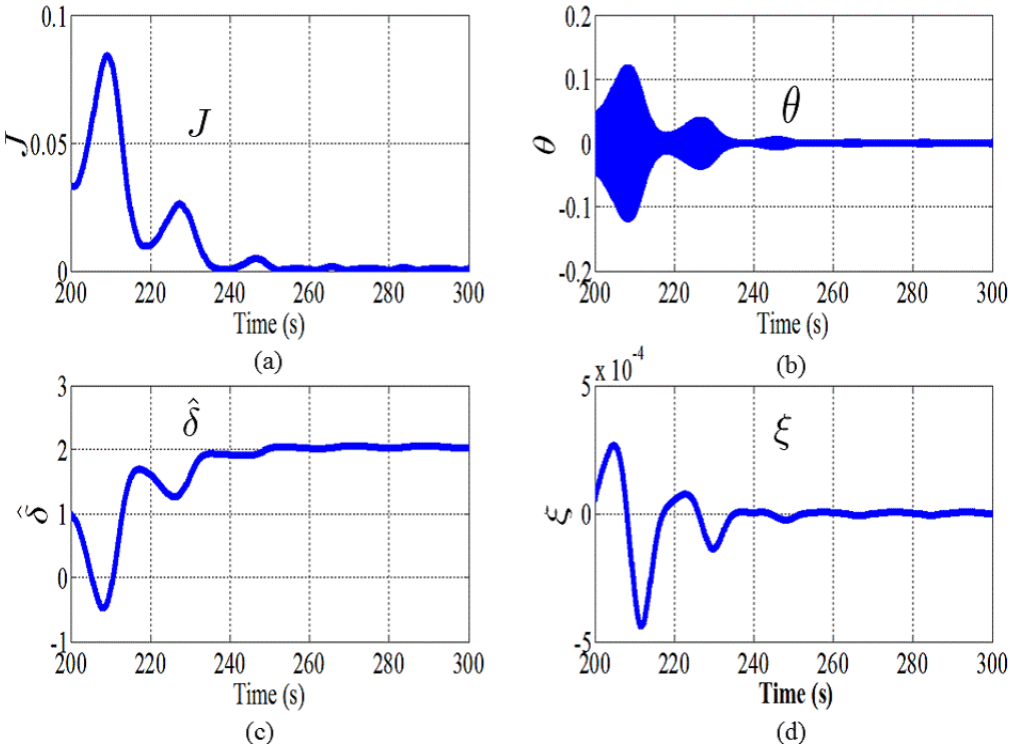


Figure 4.8: Extremum seeking result when $K = 10$ for CL-MOESP model. Time response of (a) cost function (b) the angle of the body (c) tuned parameter (d) estimated gradient of the cost function

function N. We select six variations of period of the perturbation signal which is $N = 180$ that consist of $L = 1N$, $5N$, $10N$, $20N$, $30N$, and $100N$ for analysis. It can be seen comparison period of perturbation signal L with the optimizer gain $K = 3$ and phase $\varphi = 100$ for the CL-MOESP model in Fig. 4.16. The output θ with variation period of perturbation signal L can be shown in Fig. 4.17. The best performance for variation L of simulation results of tuning dead-zone compensation by ES is $L = 10N$.

- *Period of MOPP Filter L*
  For analysis of period of the MOPP filter, it has same with period of perturbation signal.

- *Amplitude of perturbation signal a*
  To design the amplitude of the perturbation signal, we select a value which is smaller than the cost function value. For analysis, we choose the amplitude of the perturbation as $a = 0.1134$ from the cost function $J = 0.1134$ when the setting dead-zone $d = 2$ and without dead-zone compensation also with the optimizer gain $K = 3$, period of the perturbation $L = 1800$ and phase $\varphi = 100$. The variations of a for analysis are $0.1a$, $0.3a$, $0.5a$, $0.55a$, a and $10a$. The simulation results of tuning dead-zone compensation by ES with comparison the amplitude a of the perturbation signal for the CL-MOESP model represented in Fig. 4.18 and Fig. 4.19. It can be seen in Fig. 4.18 that

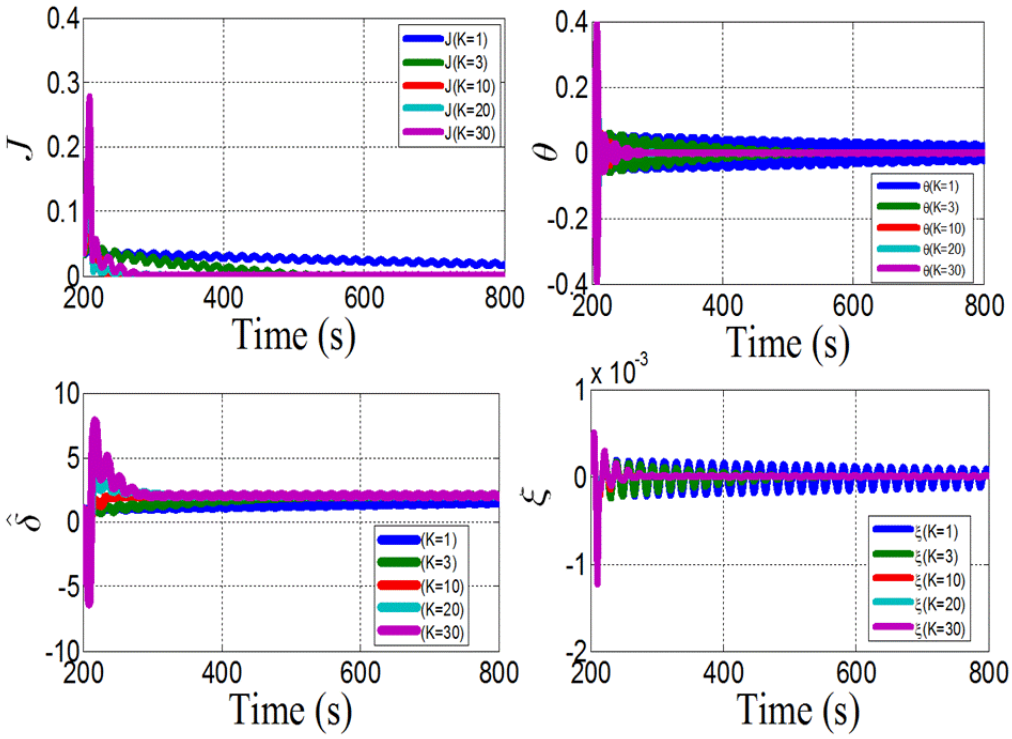


Figure 4.9: Comparison of optimizer gain K for performance system of CL-MOESP model

it is not good result when we choose amplitude of perturbation signal with high value $10a$. Hence, we should choose a value less than $1a$. The value $0.55a$ is best because the cost function after 500 second approaches zero as in Fig. 4.20 and estimation $\hat{\delta}$ approaches an appropriate value around 550 second as in Fig. 4.21.

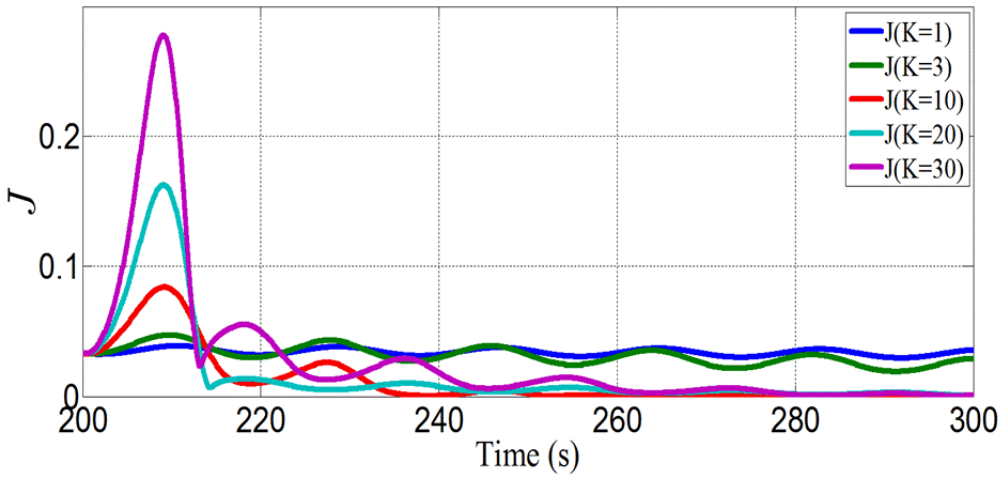


Figure 4.10: Comparison of optimizer gain K for performance of cost function J of CL-MOESP model

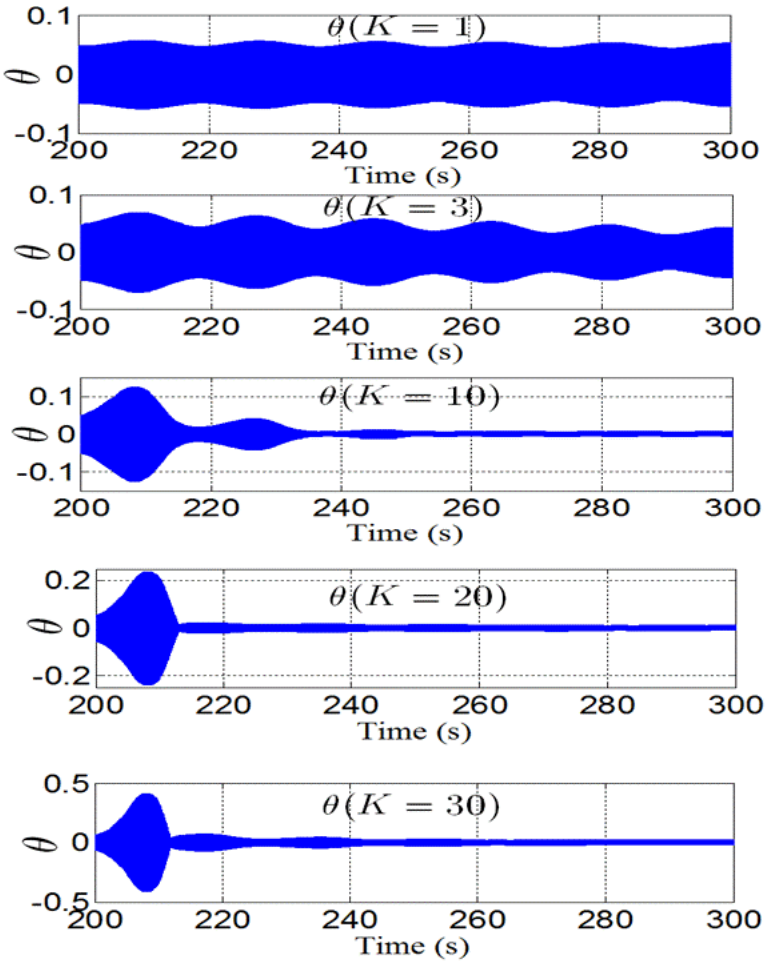


Figure 4.11: Comparison of optimizer gain K for performance of output θ of CL-MOESP model

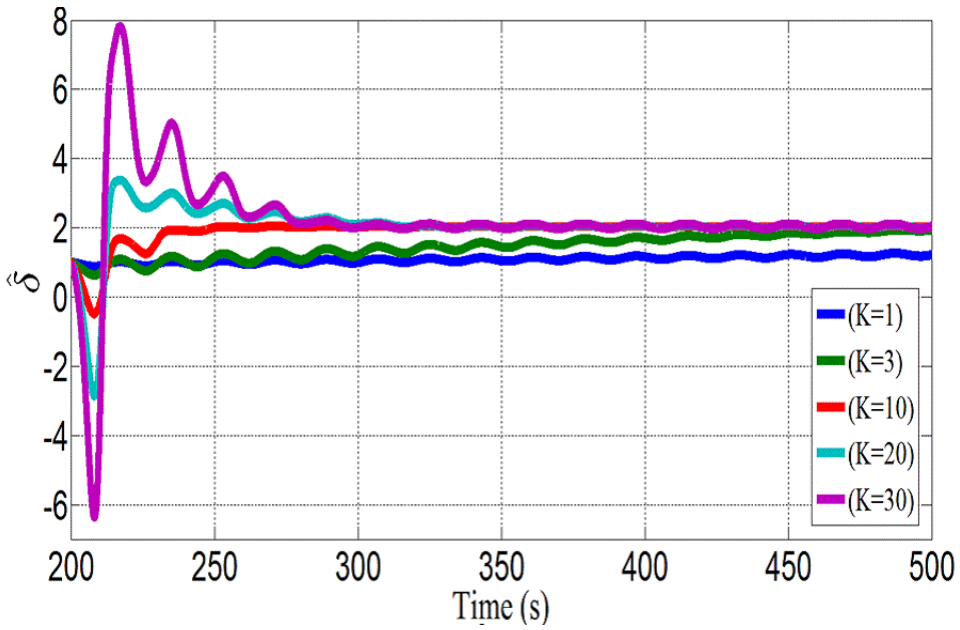


Figure 4.12: Comparison gain optimizer K for performance of estimation delta $\hat{\delta}$ of CL-MOESP model

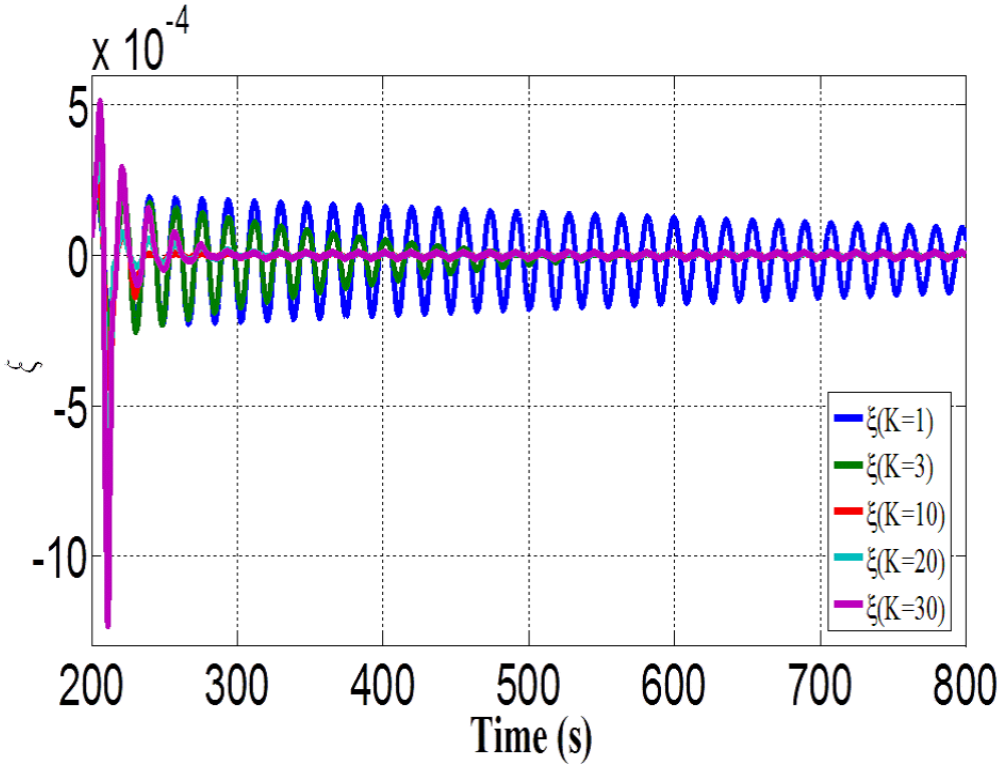


Figure 4.13: Comparison gain optimizer K for performance of ξ of CL-MOESP model

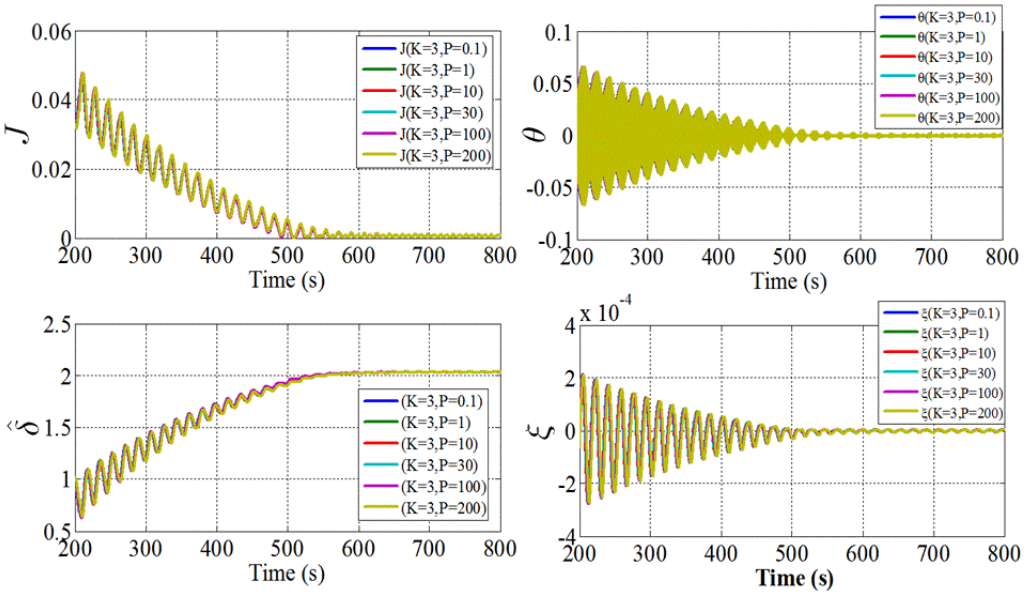


Figure 4.14: Comparison of different phases of perturbation signal φ with the gain optimizer $K = 3, (P = \varphi)$

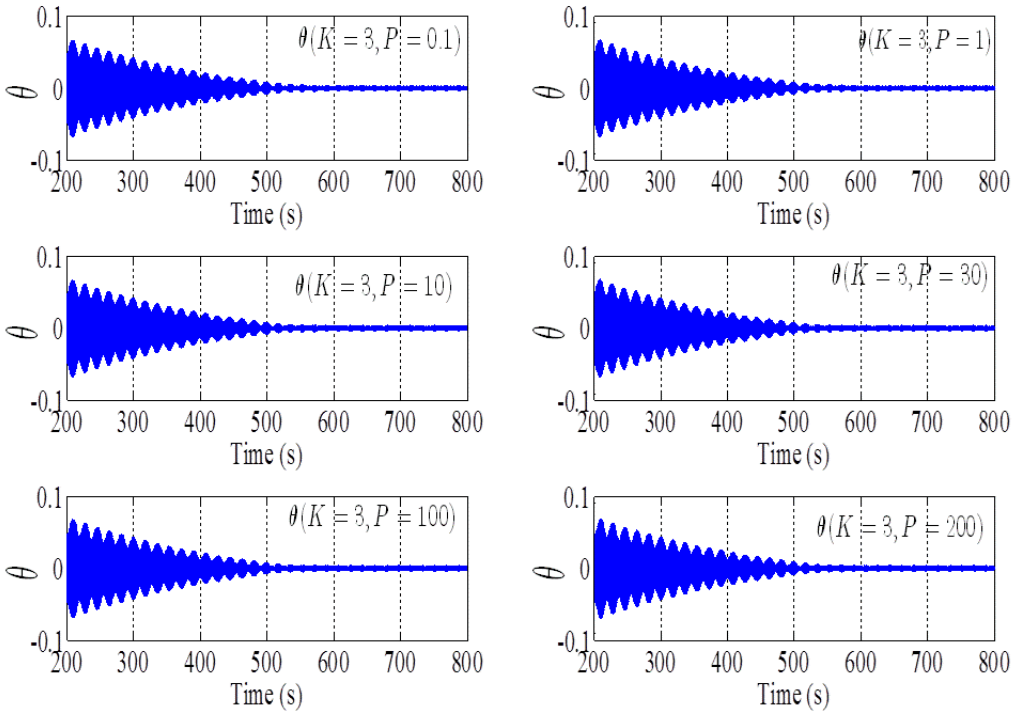


Figure 4.15: Comparison of the angle of the body θ with respect to variation of phase of the perturbation signal φ with gain optimizer $K = 3, (P = \varphi)$

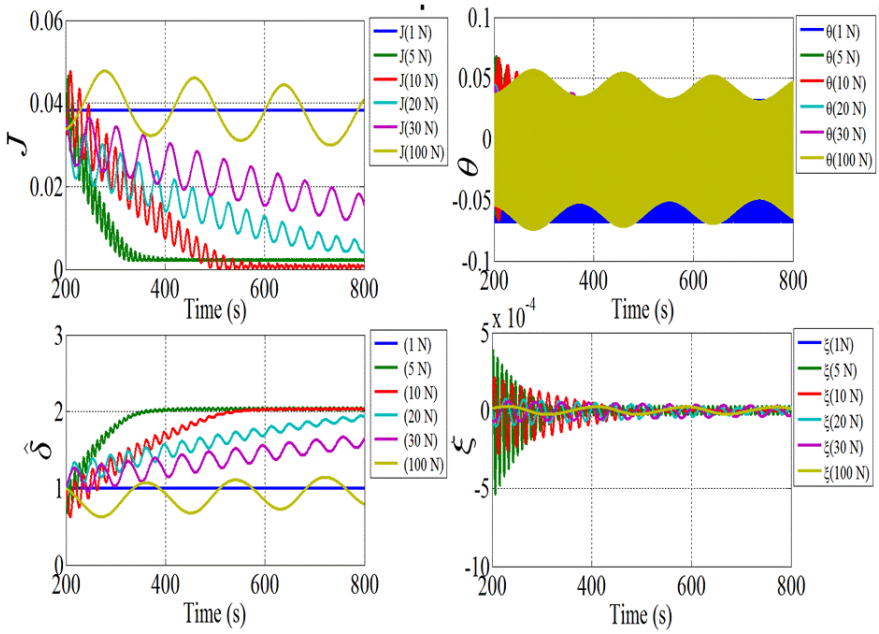


Figure 4.16: Comparison of the period L of the perturbation signal with phase $\varphi = 100$ and the gain optimizer $K = 3$

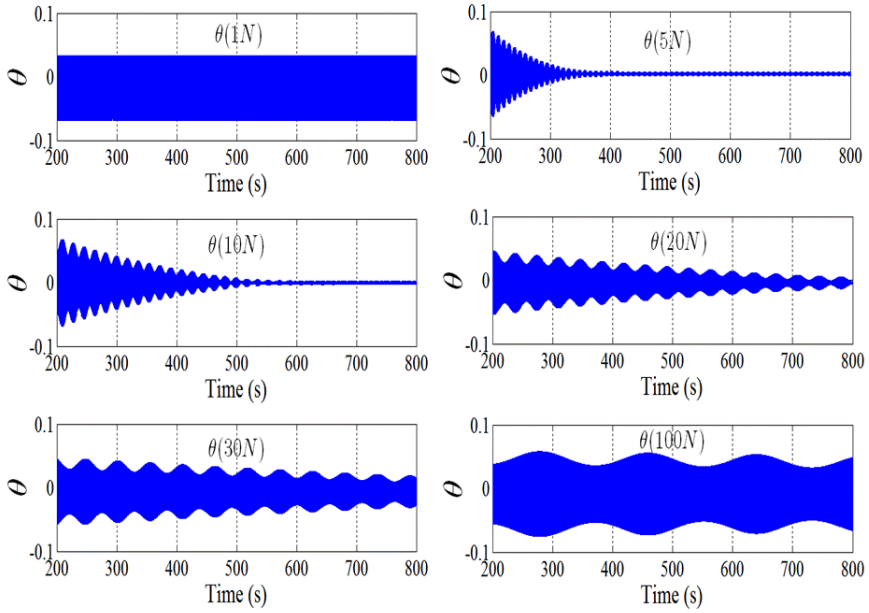


Figure 4.17: Comparison of the angle of the body θ with respect to the variation the period L of perturbation signal with phase $\varphi = 100$ and the gain optimizer $K = 3$

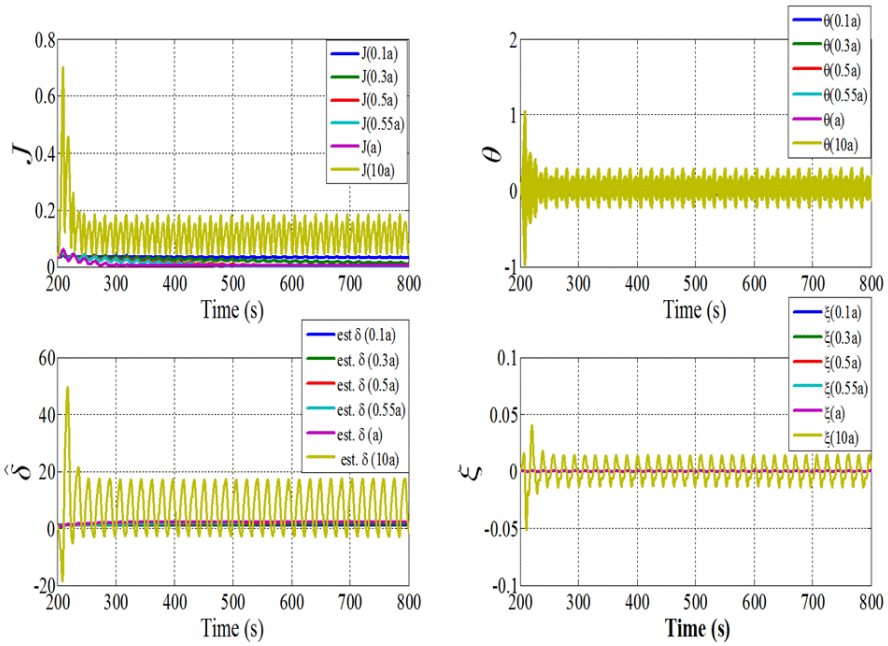


Figure 4.18: Comparison of the amplitude a of perturbation signal with period of perturbation $L = 1800$, phase $\varphi = 100$ and the gain optimizer $K = 3$

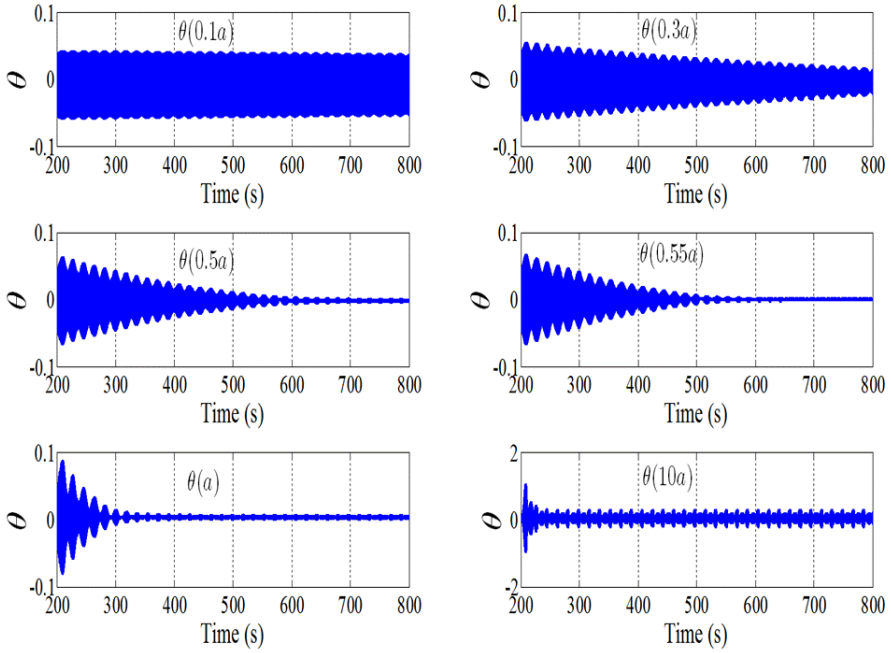


Figure 4.19: Comparison of the angle of the body θ with respect to the variation of the amplitude a of perturbation signal with period of perturbation $L = 1800$, phase $\varphi = 100$ and the gain optimizer $K = 3$

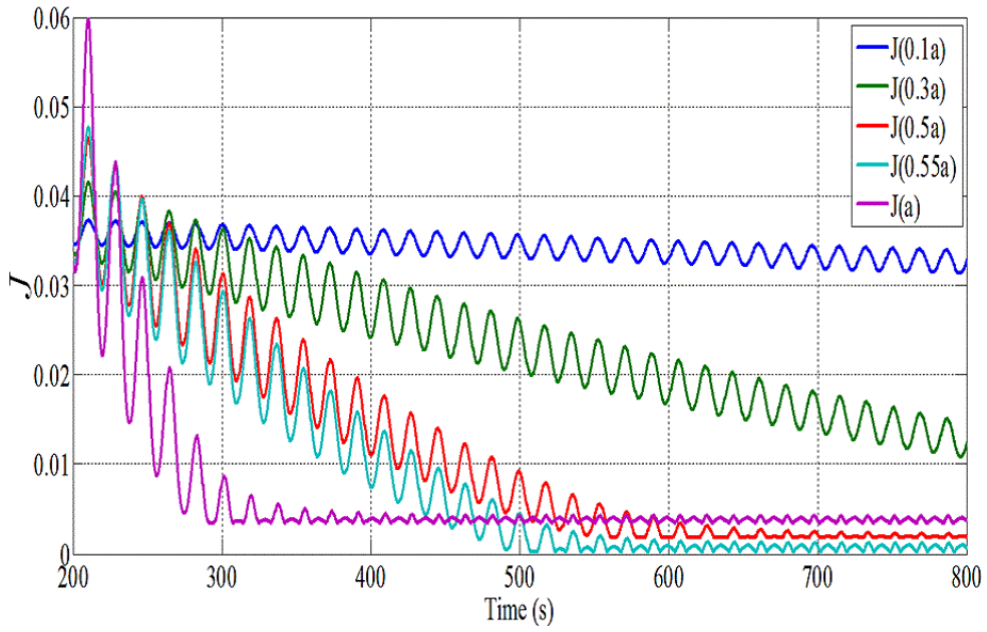


Figure 4.20: Comparison of the cost function J of amplitude a of the perturbation signal

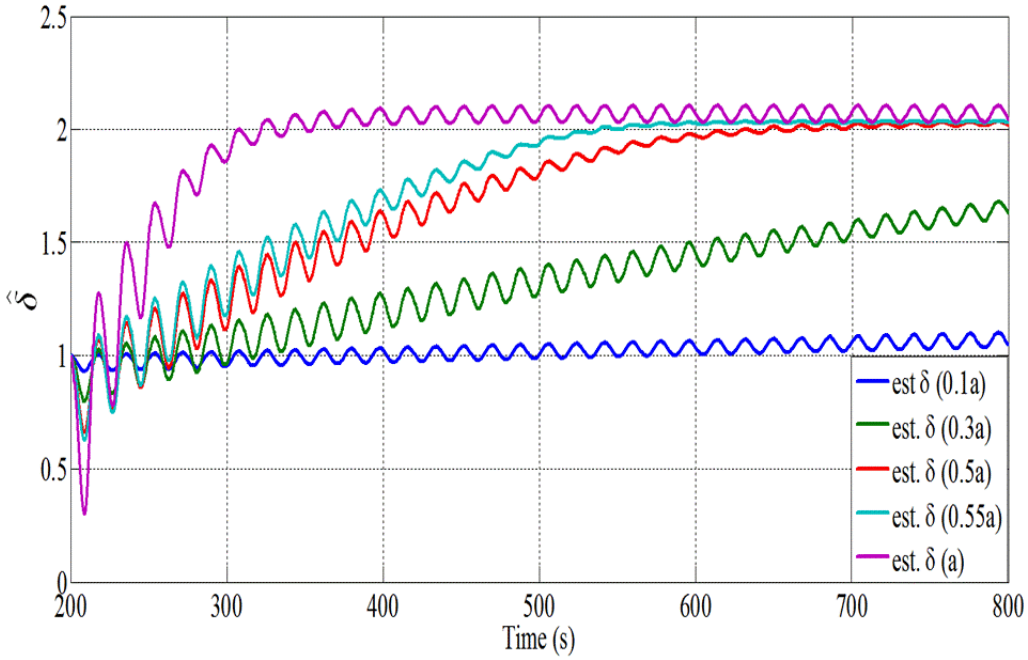


Figure 4.21: Comparison of estimation $\hat{\delta}$ of dead-zone parameter with respect to amplitude a of the perturbation signal

Chapter 5

Conclusions

This dissertation describes a new techniques to compensate the dead-zone property in real time. Some conclusions as well as shortcomings and future works are presented.

5.1 Concluding Remarks

We conclude the dissertation as follows:

- In this dissertation, we proposed discrete-time extremum seeking control by the moving average filter to tune input dead-zone compensation in real time and applied it to the stabilized self-balancing robot with the dead-zone compensation.
- The effectiveness of the proposed method is illustrated by numerical simulations. In the simulations, the compensation parameter converges to the optimal value minimizing the cost function of the performance output.

5.2 Future Works

We will evaluate the proposed discrete-time extremum seeking control for eliminating dead-zone by an experiment using e-nuvo WHEEL. However, it is not easy to apply in real time by experiment because the running of experiment e-nuvo WHEEL is quickly while the process of tuning dead-zone parameters by computer needs time also transfer data from computer to e-nuvo WHEEL has time-delay. Therefore, we need to develop a method to rapidly process tuning dead-zone parameters through micro-controllerin e-nuvo WHEEL.

Appendix A

Model of Self-balancing Robot derived from physical equations

A.1 Modelling of Self-balancing Robot as Inverted Pendulum

The equation of motion of the self-balancing robot as an inverted pendulum can be described as

$$\begin{aligned}[(M+m)r_t^2 + J_t + mlr_t\cos\theta(t) + iJ_m]\ddot{\theta}(t) - mlr_t(\sin\theta(t))\dot{\theta}^2(t) \\ +[(M+m)r_t^2 + J_t + i^2J_m]\ddot{\varphi} + c\dot{\varphi} = \eta\, i\, k_t\, u(t).\end{aligned} \tag{A.1}$$

$$\begin{aligned}[(M+m)r_t^2 + J_t + 2mlr_t\cos\theta(t) + ml^2 + J_p + J_m]\ddot{\theta}(t) - mlr_t\sin\theta(t)\dot{\theta}^2(t) \\ -mgl\sin\theta(t) + [(M+m)r_t^2 + J_t + mlr_t\cos\theta(t) + iJ_m]\ddot{\varphi} = 0\end{aligned} \tag{A.2}$$

where physical parameters are defined in Table 4.1. When we assume that

$$\sin\theta(t) \approx \theta,\ \cos\theta(t) \approx 1,\ \dot{\theta}^2(t) \approx 0, \tag{A.3}$$

(A.1) and (A.2) can be rewritten by

$$\begin{aligned}[(M+m)r_t^2 + J_t + iJ_m + mlr_t]\ddot{\theta}(t) \\ +[(M+m)r_t^2 + J_t + i^2J_m]\ddot{\varphi}(t) + c\dot{\varphi}(t) = \eta\, i\, k_t\, u(t)\end{aligned} \tag{A.4}$$

$$\begin{aligned}[(M+m)r_t^2 + J_t + 2mlr_t + ml^2 + J_p + J_m]\ddot{\theta} \\ -mgl\theta + [(M+m)r_t^2 + J_t + mlr_t + iJ_m]\ddot{\varphi} = 0\end{aligned} \tag{A.5}$$

These two equations are combined into

$$E\begin{bmatrix}\ddot{\theta}\\ \ddot{\varphi}\end{bmatrix} + F\begin{bmatrix}\dot{\theta}\\ \dot{\varphi}\end{bmatrix} + G\begin{bmatrix}\theta\\ \varphi\end{bmatrix} = \zeta u, \tag{A.6}$$

where

$$E = \begin{bmatrix}e_{11} & e_{12}\\ e_{21} & e_{22}\end{bmatrix} + ((M+m)r_t^2 + J_t)I_2,\ F = \begin{bmatrix}0 & c\\ 0 & 0\end{bmatrix},\ G = \begin{bmatrix}0 & 0\\ -mgl & 0\end{bmatrix},\ \zeta = \begin{bmatrix}\eta\, i\, k_t\\ 0\end{bmatrix}$$

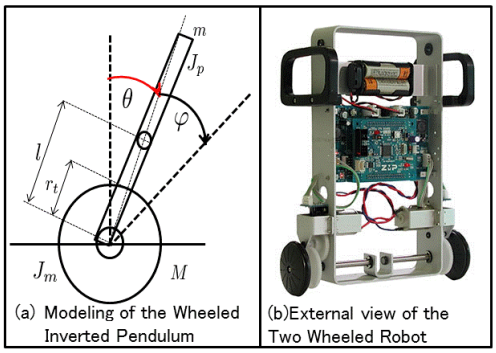


Figure A.1: Modeling of the Self-balancing robot

$$e_{11} = e_{22} = mlr_t + iJ_m \tag{A.7}$$
$$e_{12} = i^2 J_m \tag{A.8}$$
$$e_{21} = 2mlr_t + ml^2 + J_p + J_m\,. \tag{A.9}$$

From this second order differential equation, we have the state space model as

$$\dot{x} = A_c x(t) + B_c u(t) \tag{A.10}$$
$$y(t) = C_c x(t) \tag{A.11}$$

where

$$A_c = \begin{bmatrix} 0_{2\times2} & I_{2\times2} \\ -E^{-1}G & -E^{-1}F \end{bmatrix},\ x = \begin{bmatrix} \theta \\ \varphi \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix},\ B_c = \begin{bmatrix} 0_{2\times2} \\ E^{-1}\zeta \end{bmatrix}$$

$$C_c = \begin{bmatrix} \theta \\ \varphi \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix}.$$

Appendix B

CL-MOESP Identification

B.1 CL-MOESP Procedure

Subspace model identification based on Subspace Multivariable Output-Error type State-sPace closed-loop subspace model identification (CL-MOESP)is a powerful identification method to obtain a multi-input multi-output model. CL-MOESP was proposed by Hiroshi Oku, 2008. In CL-MOESP external exitation signals are used, and it provides a state-space model based on the one-shot QR factorization.

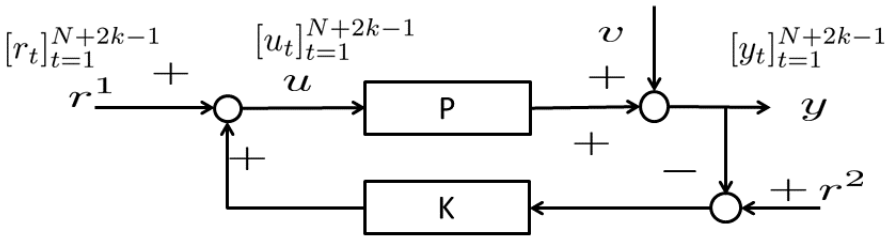
When we are given sampled data sequences $\{r_t\}_{t=1}^{N+2k-1}$, $\{u_t\}_{t=1}^{N+2k-1}$, $\{y_t\}_{t=1}^{N+2k-1}$. A closed-loop system to be identified is depicted as Fig. B.1. In Fig. B.1, P is a discrete-time linear time-invariant system, K is a stabilizing feedback controller. The external signal v is assumed to be unknown. In addition, r^1 is assumed to be an excitation signal with a sufficient persistence of excitation property, and r^2 a constant reference signal. We design block Hankel. For a given sampled data sequence u_i, the block Hankel matrix of s block rows, $\mathscr{U}_{i,j}$ is defined as

$$\mathscr{U}_{i,j} = \begin{bmatrix} u_i & u_{i+1} & \dots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & \dots & u_{i+j} \\ \vdots & \vdots & & \vdots \\ u_{i+s-1} & u_{i+s} & \dots & u_{i+j+s-2} \end{bmatrix},$$

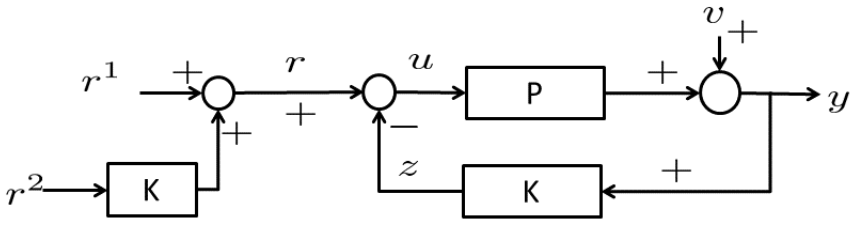
where the subscript i denotes the first element of the first column and the subscript j represents the number of columns of $\mathscr{U}_{i,j}$. The number of block row is specified by a user so that u is greater than the orders of P and $(I + KP)^{-1}$.

The procedure of CL-MOESP is given as follows.
1. Execute the QR factorization as

$$\begin{bmatrix} \mathscr{R}_{1,N} \\ \mathscr{R}_{k+1,N} \\ \mathscr{U}_{1,N} \\ \mathscr{U}_{k+1,N} \\ \mathscr{Y}_{k+1,N} \end{bmatrix} = \begin{bmatrix} L_{11} & & & & \\ L_{21} & L_{22} & & & \\ L_{31} & L_{32} & L_{33} & & \\ L_{41} & L_{42} & L_{43} & L_{44} & \\ L_{51} & L_{52} & L_{53} & L_{54} & L_{55} \end{bmatrix} \begin{bmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \\ Q_4^T \\ Q_5^T \end{bmatrix}$$



(a) A closed-loop system



(b) A closed-loop system equivalent

Figure B.1: A closed-loop system for CL-MOESP

2. By defining

$$\begin{aligned} L_3 &:= \begin{bmatrix} L_{31} & L_{32} \end{bmatrix} \\ L_4 &:= \begin{bmatrix} L_{41}^T \\ L_{42}^T \end{bmatrix} \\ L_5 &:= \begin{bmatrix} L_{51} & L_{52} \end{bmatrix}, \end{aligned}$$

compute $\Upsilon^{\frac{1}{2}}$ as follows

$$\begin{aligned} \Upsilon^{\frac{1}{2}} &= L_5 P^T \left(PP^T\right)^{-\frac{1}{2}} \\ P &:= L_3 - L_3 L_4^T (L_4 L_{41}^T)^{-1} L_4 \,. \end{aligned}$$

3. Execute SVD of $\Upsilon^{\frac{1}{2}}$

$$\Upsilon^{\frac{1}{2}} = \begin{bmatrix} U & U^\perp \end{bmatrix} \begin{bmatrix} \Sigma & \\ & \tilde{\Sigma} \end{bmatrix} \begin{bmatrix} V \\ V^\perp \end{bmatrix}^T$$

to estimate the extended observability matrix of P. The diagonal matrix $\Sigma \in \Re^{n \times n}$ has n dominant singular values as its diagonal entries , and n corresponds to the order of P.

4. An estimate of the quadruple of (A, B, C, D) of a state-space representation of P can be obtained as follows.

(A, C) are estimated by

$$\begin{aligned}\hat{A} &= U^{\dagger}(1:(k-1)\ell,:)U(\ell+1:k\ell,:)\\ \hat{C} &= U(1:\ell,:)\,.\end{aligned}$$

To estimate (B, D), define

$$\begin{aligned}[\alpha_1\ \alpha_2\ \ldots\ \alpha_k] &:= (U^{\perp})^T\\ [\beta_1\ \beta_2\ \ldots\ \beta_k] &:= (U^{\perp})^T L_5 L_4^T (L_4 L_4^T)^{-1}\\ U_1 &:= U(1:(k-1)\ell,:)\end{aligned}$$

$$\begin{bmatrix}\hat{B}\\ \hat{D}\end{bmatrix} = \begin{bmatrix}0 & U_1\\ I_\ell & 0\end{bmatrix}^{\dagger} \begin{bmatrix}\alpha_1 & \alpha_2 & \ldots & \alpha_k\\ \alpha_2 & & \alpha_k & \\ \vdots & \alpha_k & & \\ \alpha_k & & & 0\end{bmatrix}^{\dagger} \begin{bmatrix}\beta_1\\ \beta_2\\ \vdots\\ \beta_k\end{bmatrix}.$$

Appendix C

Stability Analysis Theorems

C.1 Averaging Theorem

The averaging theorem (Theorem 8.3 in Khalil [17]) is as follows.
Consider the singularly perturbed system

$$\dot{x} = f(t, x, z, \epsilon)$$
$$\epsilon\dot{z} = g(t, x, z, \epsilon).$$

Assume that the following assumptions are satisfied for all $(t, x, \epsilon) \in [0, \infty] \times B_r \times [0, \epsilon_0]$ for some domain $B_r \subset R^n$.

- $f(t, 0, 0, \epsilon) = 0$ and $g(t, 0, 0, \epsilon) = 0$. That is, $(x, z) = (0, 0)$ is an equilibrium point.
- The equation $0 = g(t, x, z, 0)$ has an isolated root $z = h(t, x)$ satisfing $h(t, 0) = 0$.
- The functions f, g, and h and their partial derivatives up to order 2 are bounded for $z - h(t, x) \in B\rho$.
- The origin of the reduced system

  $$\dot{x} = f(t, x, h(t, x), 0)$$

  is exponentially stable.
- The origin of the boundary-layer system

  $$\frac{dy}{d\tau} = g(t, x, y + h(t, x), 0)$$

  is exponentially stable uniformly in (t, x).

Then there exists $\epsilon^* > 0$ such that, for all $\epsilon^* > \epsilon > 0$, the origin of the singularly perturbed system is exponentially stable.

Appendix D

Control Blocks Simulink and MATLAB Programs

D.1 Control Blocks Simulink of Two-Wheeled Robot Model

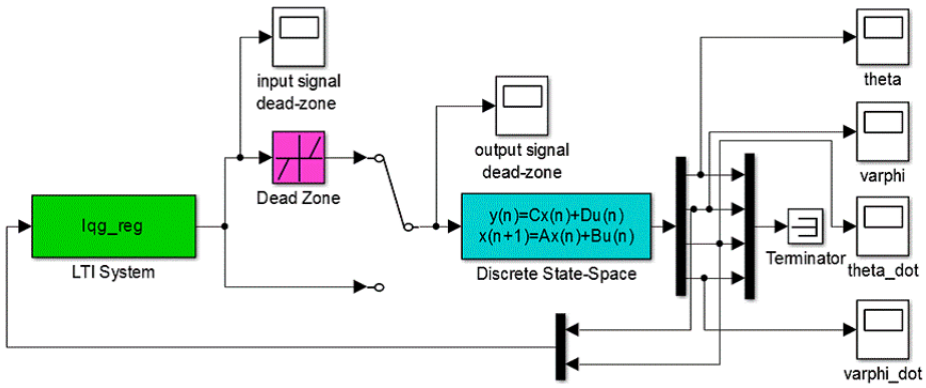


Figure D.1: Closed-loop system with the LQG controller, the dead-zone and the plant block

D.2 MATLAB Programs

We describe the MATLAB code to obtain the discrete-time LQG controller for derived physical equations of the self-balancing robot model

clear all

clc

%Parameter enuvo wheel

%————————————————————

$m = 0.5392; M = 0.071; Jp = 2.160 * 10^{-}3; Jt = 8.632 * 10^{-}5;$

$Jm = 1.30 * 10^{-}7; l = 0.1073; rt = 0.02485; c = 1 * 10^{-}4;$

$I = 30; Kt = 2.79 * 10^{-}3; eta = 0.75; g = 9.80665;$

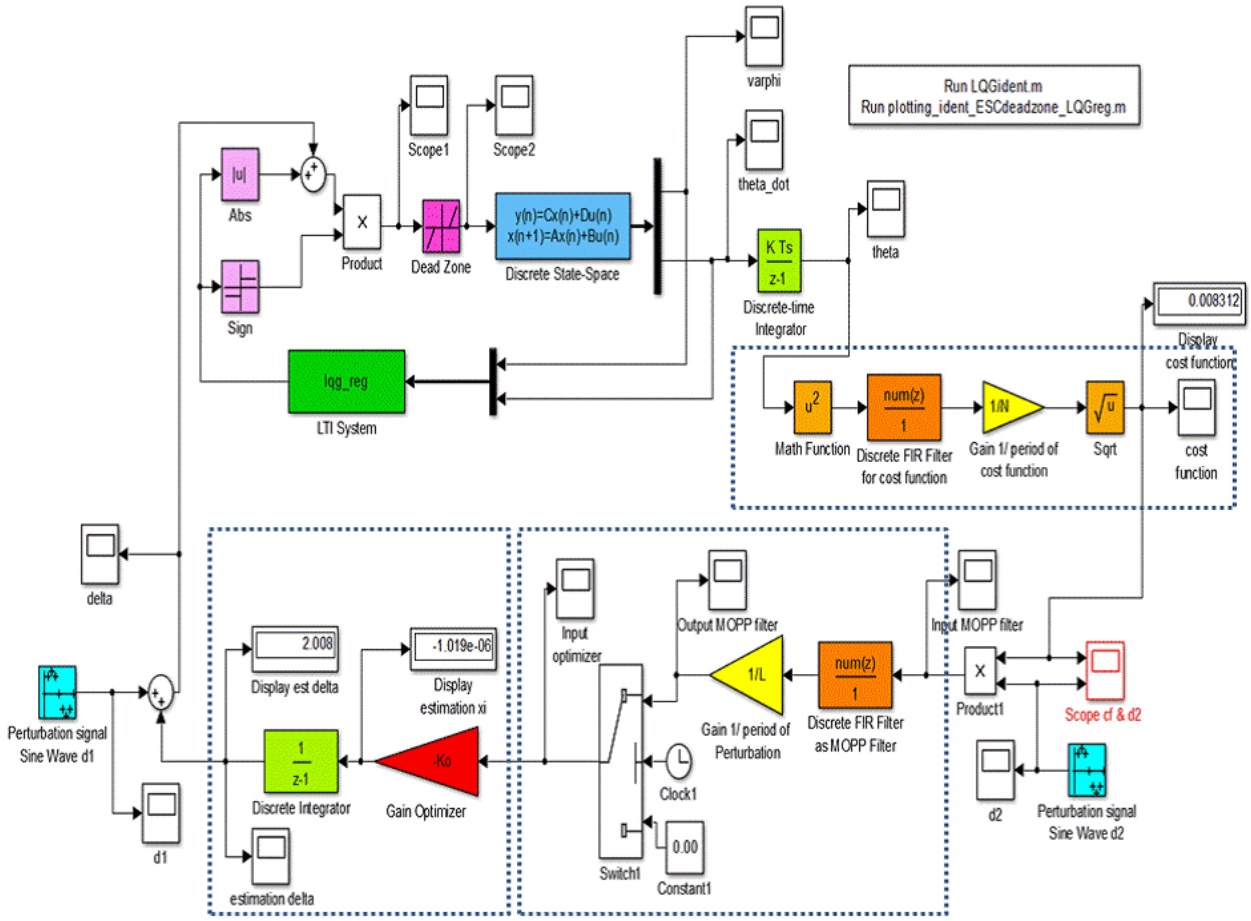


Figure D.2: Discrete-time extremum seeking control block

$Mmrt2 = (M + m) * rt^2; mlrt = m * l * rt; Js = Jp + Jt + Jm;$
%————————————————————
%step1
$atheta = Mmrt2 + mlrt + Jt + I * Jm;$
$aphi = Mmrt2 + Jt + I^2 * Jm;$
$btheta = Mmrt2 + 2 * mlrt + m * l^2 + Js;$
$bphi = Mmrt2 + mlrt + Jt + I * Jm;$
%————————————————————
%step2
$Mm = [atheta\ aphi; btheta\ bphi];$
$invMm = inv(Mm);$
$Dm = [0\ \ c; 0\ \ 0];$
$Km = [0\ \ 0; -m * g * l\ \ 0];$
%————————————————————
%step3 State Space Model
$A = [zeros(2,2)\ \ eye(2); -invMm * Km\ \ -invMm * Dm];$
$B = [0; 0; -invMm * [eta * I * Kt; 0]]$
$C = [0\ \ 1\ \ 0\ \ 0; 0\ \ 0\ \ 1\ \ 0];$
$D = [0; 0];$

$G = ss(A, B, C, D)$;
% Discrete state space model
$Gd = c2d(G, 0.01, 'zoh')$;
%stabilizable
$checkeigenGd = abs(eig(Gd))$;
% Retrieve the matrices
$[Ad, Bd, Cd, Dd] = ssdata(Gd)$;
%————————————————————————
%step 4
%Get discrete LQR Servo Controller
$Q = eye(size(Gd.A))$;
$R = 1$;
[K,S,e] =dlqr(Ad, Bd, Q, R);
$checkeigen\, K = abs(eig(Ad - Bd * K))$;
%————————————————————————
%step 5
% Compute the Kalman filter gains
% Assume rms noise of 1% on each sensor channel
$Rv = 0.01^2 * eye(2)$;
% Input Rw
$Rw = 1$;
$sensors = [1, 2]$; % d and y are sensed
$known = [1]$; % force u
$P = ss(Ad, [Bd\ \ Bd], Cd, [Dd\ \ Dd], 0.01)$;
[Observer, Ko] = kalman(P, Rw, Rv, [], sensors, known);
%step 6
%————————————————————————————————
% Create the regulator and the closed-loop system
% $lqgregulator = lqgreg(Observer, K)$
$feedin = [1]$; %force u
$feedout = [1, 2]$; %d and y
$Gcl = feedback(Gd, lqgregulator, feedin, feedout, +1)$;
%————————————————————————————————-
% Compute and plot the initial condition response
% Set $x_1(0) = 0.1$, all others to zero.
$x0 = zeros(8, 1)$;
$x0(1) = 0.1$;
$figure(1), clf$;
$initial(Gcl, x0)$;
% Plot all states and compare actual values with Kalman estimates;
[y, t, x] = initial (Gcl, x0, 10);
figure(2), clf;
subplot(2, 2, 1), stairs($t, x(:, [15])$), grid, legend($'x_1 = theta', 'x_1hat', 0$);
title('Response of states and predictive estimates to $x_1(0) = 0.1$');
xlabel('Time (s)');
subplot(2, 2, 2), stairs($t, x(:, [26])$), grid, legend($'x_2 = Phi', 'x_2hat', 0$);

```
xlabel('Time (s)');
subplot(2, 2, 3), stairs(t, x(:, [37])), grid, legend('x_3 = thetadot','x_3hat', 0);
xlabel('Time (s)');
subplot(2, 2, 4), stairs(t, x(:, [48])), grid, legend('x_4 = Phidot','x_4hat', 0);
xlabel('Time (s)');
```

Publications

Journal papers

[1] D. Novita and S. Yamamoto: Extremum Seeking for Dead-Zone Compensation and Its Application to a Two-Wheeled Robot, *Journal of Automation and Control Engineering*, Vol. 3, No. 3, June 2015, (ISSN: 2301-3702, DOI:10.12720/joace).

Conference papers

[1] D. Novita and S. Yamamoto: Extremum Seeking for Dead-Zone Compensation and Its Application to a Two-Wheeled Robot, Proceeding of the 2nd International Conference on Control, Robotics and Cybernetics (ICCRC) August 8th -10th , 2014, Singapore.

Bibliography

[1] K. B. Ariyur and M. Krstić, "Analysis and Design of Multivariable Extremum Seeking," in *Proceedings of The American Control Conference*, Anchorage, AK, May 8-10, 2002, pp. 2903–2908.

[2] K. Ariyur and M. Krstić, *Real-time optimization by Extremum-seeking Control*. Wiley-Interscience, 2003.

[3] W. M. Bessa, M. S. Dutra, and E. Kreuzer, "Sliding mode control with adaptive fuzzy dead-zone compensation of an electro-hydraulic servo-system," *Journal of Intelligent and Robotic Systems*, vol. 58, no. 1, pp. 3–16, May, 2009.

[4] ——, "An adaptive fuzzy dead-zone compensation scheme and its application to electro-hydraulic systems," *Journal of the Brazilian Society of Mechanical Scientist and Engineering*, vol. XXXII, no. 1, pp. 1–7, January-March, 2010.

[5] M. Chan and K. Kong, "Automatic Controller Gain Tuning of a Multiple Joint Robot Based on Modified Extremum Seeking Control," in *Proceedings of The 18th International Federation of Automatic Control (FAC) World Congress*, Milano (Italy), August 28 - September 2, 2011, pp. 4131–4136.

[6] J.-Y. Choi, M. Krstić, K. B. Ariyur, and J. S.Lee, "Extremum Seeking Control for Discrete-Time Systems," *Journal of IEEE Transactions on Automatic Control*, vol. 47, no. 2, pp. 318–323, 2002.

[7] P. Frihauf, M. Krstić, and T. Bas, "Finite-Horizon LQ Control for Unknown Discrete-Time Linear Systems via Extremum Seeking," in *Proceedings of The 51st IEEE Conference on Decision and Control*, Maui, Hawaii (USA), December 10-13, 2012, pp. 5717–5722.

[8] M. Haring, N. V. de Wouw, and D. Nešić, "Extremum-seeking control for nonlinear systems with periodic steady-state outputs," *Automatica*, vol. 49, no. 6, pp. 1883–1891, June 2013.

[9] M. Haring, "Extremum-seeking control for steady-state performance optimization of nonlinear plants with periodic steady-state outputs," Ph.D. dissertation, Eindhoven University of Technology, June, 2011.

[10] H.Oku, "Identification experiment of a cart system using closed-loop subspace model identification methods," in *Proceedings of SICE Annu. Conf. on Control Systems,(in Japanese)*, 2008.

[11] ——, "On asymptotic properties of moesp-type closed-loop subspace model identification," in *Proceedings of the 19th International Symposium on Mathematical Theory of Networks and Systems*, Budapest, Hungary, July 5-9, 2010, pp. 1015–1022.

[12] B. Hunnekens, M. Haring, N. V. D. Wouw, and H. Nijmeijer, "Steady-state performance optimization for variable-gain motion control using extremum seeking," in *Proceedings of The 51st IEEE Conference on Decision and Control (CDC)*, Maui, Hawaii (USA), December, 10-13, 2012, pp. 3796–3801.

[13] J. F. Franklin, J. D. Powell and M. L. Workman, *Digital Control of Dynamic Systems*, 3rd ed. Addison Wesley Longman, Inc, 1998.

[14] J. O. Jang, H. T. Chung, and G. J. Jeon, "Saturation and deadzone compensation of systems using neural network and fuzzy logic," in *Proceedings of The American Control Conference*, Potland, OR (USA), June 8-10, 2005, pp. 1715–1720.

[15] N. J. Killingsworth and M. Krstić, "Auto-tuning of PID controllers via extremum seeking," in *Proceedings of The American Control Conference*, Potland, OR (USA), June 8-10, 2005, pp. 2251–2256.

[16] ——, "PID Tuning Using Extremum Seeking," *IEEE Control Systems Magazine*, pp. 70–79, February, 2006.

[17] H. K.Khalil, *Nonlinear Systems*, 1st ed. Macmillan Publishing Company, 1992.

[18] K. Kong, K. Inaba, and M. Tomizuka, "Real-Time Nonlinear Programming by Amplitude Modulation," in *Proceedings of ASME Dynamic Systems and Control Conference*, Ann Arbor, Michigan (USA), October 20-22, 2008, pp. 1–8.

[19] M. Krstić and H.-h. Wang, "Stability of extremum seeking feedback for general nonlinear dynamic systems," *Automatica*, vol. 36, pp. 595–601, 2000.

[20] J. D. J. Rubio, Z. Zamudio, J. Pacheco, and D. M. Vargas, "Proportional Derivative Control with Inverse Dead-Zone for Pendulum Systems," *Journal of Mathematical Problems in Engineering*, pp. 1–9, 2013.

[21] A. Scheinker, "Extremum Seeking for Stabilization," Ph.D. dissertation, University of California, San Diego, 2012.

[22] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control Analysis and design*, 2nd ed. John Wiley and Sons, 2001.

[23] Y. Tan, W. Moase, and C. Manzie, "Extremum seeking from 1922 to 2010," in *Proceedings of The 29th Chinese Control Conference*, Beijing (China), December 10-13, 2010, pp. 14–26.

[24] G. Tao and P. V. Kokotović, "Adaptive control of plants with unknown dead-zones," *Journal of IEEE Transactions on Automatic Control*, vol. 39, no. 1, pp. 59–68, January, 1994.

[25] A. R. Teel, J. Peuteman, and D. Aeyels, "Semi-global practical asymptotic stability and averaging," *Journal of Systems and Control Letters*, vol. 37, pp. 329–334, May, 1999.

[26] Y. Wakasa and K. Tanaka, "FRIT for Systems with Dead-Zone and Its Application to Ultrasonic Motors," *Journal of IEEJ Transactions on Electronics, Information and Systems*, vol. 131, no. 6, pp. 1209–1216, 2011.

[27] ZMP, *Stabilization and Control stable running of the Wheeled Inverted Pendulum Development of Educational Wheeled Inverted Robot e-nuvo WHEEL ver. 1.0*. ZMP INC, (in Japanese). [Online]. Available: {http://www.zmp.co.jp/e-nuvo/pdf/wheel/wheel_paper.pdf}

[28] ——, *Manual Designing control system Robot e-nuvo WHEEL ver. 1.0.0"*. ZMP INC, (in Japanese), 2007.